Information Flow in Neural Circuits

Submitted in partial fulfillment of the requirements for
the degree of
Doctor of Philosophy
in
Electrical and Computer Engineering

Praveen Venkatesh

B.Tech. (Hons.), Electrical Engineering, Indian Institute of Technology Madras
M.S., Electrical and Computer Engineering, Carnegie Mellon University

Carnegie Mellon University
Pittsburgh, PA

May 2021

Acknowledgments

I have had the immeasurably good fortune of being advised by Prof. Pulkit Grover. Throughout my Ph.D., he has been not just an advisor, but an active participant in my research. His undying optimism is extremely infectious: every meeting was sure to re-energize me and leave me feeling hopeful about whatever problem I was pursuing. He has been both hands-on and hands-off with a great deal of sensitivity, allowing me find my own way while providing me every available resource. He has also been a stellar mentor: the latter years of my Ph.D. saw us spending an innumerable number of hours carefully strategizing my future career decisions.

To say that I have learned a great deal from Pulkit would be an understatement. It would be more accurate to say that he has shaped the researcher in me, going far beyond just imparting the basic knowledge of how to do research. He has taught me how to find new problems, how to interact with collaborators, how to prioritize my tasks, and even how to procrastinate (by doing something useful)! He has helped me understand the difference between practicality and purity, and between rules and ethics. He has also shown me what sort of ideal I must strive for in my research and in its presentation. Above all, Pulkit has taught me how to be kind—to others, and also to myself. For all this, and more, I shall be eternally grateful to him.

I am also immensely grateful to my thesis committee, Profs. Todd Coleman, Marlene Behrmann, Robert Kass, José Moura and Cosma Shalizi. Todd has been a mentor and a huge supporter of my work from the very beginning. I will never forget his advice—"Follow your nose!"—and his unique ability to inspire you, even when you are at your lowest. Marlene served on my committee only for a while, but she was a huge influence in my research as a whole. Without her enthusiastic support, Pulkit and I would not have got such a firm foothold in neuroscience: she truly appreciated what we as engineers brought to the table, and took the time to teach us during our meetings. I am also extremely thankful to her for her patience, for we were not always the best collaborators to her, even though she was to us. Rob has had a hidden hand in much of my Ph.D. work: his introducing us to Dr. Mark Richardson gave rise to much of the work I did on EEG as well as Epilepsy, and my work on information flow was inspired by many discussions with him. In many ways, his inputs have guided the core problem formulation, and he has also been one of the greatest supporters of the finished product of our work on information flow. I am also extremely grateful to both Rob and Marlene for their letters of recommendation (and my endless requests thereof), which have gone a long way in helping me secure my next position. José has also been a

great supporter from behind the scenes and helped ensure that this thesis stayed grounded and practical. Cosma was extremely gracious and agreed to step in at the last minute, and took a great deal of care and effort to read my work and provide feedback. I am extremely grateful to all of my committee members for being so deeply engaged with my work and invested in my future. It is truly affirming to know that you have the backing of so many people, and helps keep you grounded and on the right path.

I have to sincerely thank Dr. Mark Richardson and Dr. Vasileios Kokkinos, for being our longest collaborators. I was given the unparalleled opportunity to sit in their weekly epilepsy conferences and learn how doctors make their decisions about localizing seizure foci and treating epilepsy: it was truly a one-of-a-kind experience. They also took a lot of time to teach us, and truly believed in our ability to provide a new perspective within medicine. I am very grateful to them for their patience, and for their unwavering support.

I would also like to thank Profs. Bobak Nazer and Venkatesh Saligrama for showing me a completely new perspective on research over one summer. They taught me what working on theory was really like, and helped me understand where I fell on the theory-practice spectrum. Their advice on crafting a research profile that can appeal to universities seeking faculty candidates helped me pick problems in my Ph.D. and is something that will stick with me for a long time.

It would be remiss of me not to mention my professors and mentors from my undergraduate days, who each left their own unique and lasting impressions on my understanding of the world. I thank my B.Tech. thesis advisor, Prof. Radhakrishna Ganti for teaching me how to work with real systems. I am grateful to Prof. Harishankar Ramachandran for being an amazing teacher, and for helping me understand how to construct simulations. I am also very grateful to Prof. Upinder Bhalla at the National Center for Biological Sciences, Bangalore, for introducing me to the world of neuroscience. I would also like to thank Profs. Nitin Chandrachoodan, Nandita Dasgupta, Krishan Jagannathan, L. Sriramkumar and Srinivasa Chakravarthy, for supporting me and giving me advice at various points in my undergraduate life.

I have had the great privilege of sharing these past many years with a wonderful group of labmates and collaborators. I am extremely grateful to each one of them: Yaoqing, for setting the bar to ever strive towards while being unfailingly modest; Haewon, for being an awesome deskmate, for all of her thoughtful feedback over the years, and for being my friend after I abandoned her on the convex optimization course; Sanghamitra, for collaborating with me on so many papers, and for being an inspiration in productivity; Alireza, for showing me how to be determined and persevere through long collaborations; Jiaming, for introducing a little fun in the B-200 wing every so often; Chaitanya, for many long research conversations that would often end up nowhere; Sara, for teaching me neuroscience and putting up with my convoluted questions; and Neil and Ariel, for reminding me what it is like to have a fresh starry-eyed enthusiasm for research, and to never forget to be that person. I will forever remember the moments we shared over many afternoons and late evenings in the B-200 wing, always full of laughter, and which would slowly grow to include the whole lab. Thank you all so much for enlivening my graduate school days!

My lab extends beyond my immediate Ph.D. cohort, of course, and I am also thankful to the entire Neural Web Group. In particular, I owe a huge debt of gratitude to Ashwati, for all of the driving lessons, for allowing me to borrow her car during the pandemic, and

for being there whenever I needed to vent. I am also indebted to the two other postdocs I worked closely with, Amanda and Sarah, both of whom have put up with me a great deal during our respective collaborations.

Much of the work I did would not have been possible without the help of various undergraduate and masters students who worked closely with me over the years. I thank Susan, Danish, Daniel, Leo, Ritesh, Ivy, Alankrita and Revanth for many productive conversations from which I learned a little about what it means to be in an advisor's shoes. I am extremely grateful to have had the opportunity to interact with them, and to have briefly had a mini-group with some of them, who were open to me exploring different advising styles and group meeting formats.

I have also been extremely fortunate to have had two fantastic collaborators in Gabe and Aditya. With Aditya I had a brief but intense collaboration, and he showed me through example how theoretical research is done. With Gabe, I have had an avid partner in exploring a completely new subject area, while learning how to be an independent researcher, as well as someone who shared many of my frustrations and joys of interdisciplinary research. Needless to say, they have also become great friends!

Outside of my lab and my close collaborators, I am also grateful to the Neurostats and Scabby groups, as well as Prof. Barbara Shinn-Cunningham's group, all of which have given me great deal of feedback on my work. In particular, I must thank Prof. Byron Yu and Prof. Steve Chase for teaching me so much about neuroscience and being extremely encouraging of my work. I must also thank Profs. Bard Ermentrout, Maysam Chamanzar, Shawn Kelley and Jeff Weldon, for many fruitful collaborations. The collaborative environment at CMU made much of my research possible, by simply creating the possibility for so many interactions. I am extremely grateful to Kara Knickerbocker, for being amazingly available, approachable and helpful with all matters in the B-200 wing. I also thank Nathan Snizaski and the rest of the ECE, OIE and CMU administrative staff, for making my life easier through so many smooth and pleasant interactions.

It goes without saying that the research would not have been possible without my funding sources: NSF CCF-1350314, NSF CNS-1702694, SRC SONIC, as well as grants from CMLH and the Chuck Noll Foundation. I have had the great honor of receiving a number of fellowships while at CMU, which helped me believe in the work I was doing. I was supported in part by a CMU Dean's Fellowship, a Henry L. Hillman Presidential Fellowship, a Dowd Fellowship and a CMLH Fellowship, at various points during my Ph.D. I am very grateful to Philip and Marsha Dowd for their financial support and encouragement.

My life at CMU was made richer by many friendships, which carried me through the ups and downs of Ph.D. life. I am grateful to Dhivya, Satwik, Bhavana and Shrimai, for many fun-filled evenings, to Shiny and Anit for being among my first friends at CMU, and to Sibi for being my first friend period. I owe a huge debt of gratitude to Vishwanath, who has been my housemate and closest friend through this long journey: from learning how to live as responsible independent adults, to having thoughtful and deep conversations about research and teaching. He has taught me to always remember that I am at my core an engineer, to keep my hobbies alive and to keep building things.

I am deeply grateful to my fiancée, Ranjani, who has been everything I could ever ask for in a partner. She has always understood me, supported me and encouraged me. Above all, she has made me a better human being.

Finally, my gratefulness towards my family knows no bounds. I am grateful to my mother, my father and my brother, for always being there for me. Mere words are insufficient to express gratitude for providing me such a full life, for giving me all of the freedom I could ever want in all things, and for having an unshakable belief in me as a person. I owe everything to them.

Praveen Venkatesh
14 May, 2021

Abstract

Neuroscience has witnessed rapid advances in the last half century, with modern neurotechnologies now allowing us to record simultaneously from hundreds, if not thousands, of neurons. It is conceivable that in the near future, we might be able to record from every single neuron in the brain, or in a subsystem—indeed, this is already the case for several small animals. Would this alone suffice to help us obtain a complete understanding of the brain? Recent experiences with artificial neural networks suggests that this is not the case: even when we can "record" from every single node of a neural circuit, make interventions and understand learning rules, we may not truly understand a system.

This calls for new theoretical and computational frameworks that are capable of providing objective explanations about how these neural circuits function. We need new tools for providing explanations at each of Marr's levels, ultimately leading to an understanding of how neural mechanisms give rise to behavior. Despite the tremendous advances in experimental neuroscience and neurotechnology, there is a considerable gap in our theoretical and computational capability to extract such an understanding.

This thesis aims to address the aforementioned theoretical gap in one narrowly focused problem domain—*information flow*. Inferring the flows of information in healthy and diseased states of the brain is essential in neuroscience because it could help us understand how the brain performs specific tasks. In particular, we require an understanding of information flow that enables us to (i) track the flows of one or more specific messages; (ii) capture how these flows evolve over time, especially in feedback systems; (iii) draw meaningful interpretations about the underlying computations, and (iv) identify interventions that can modulate flows to treat brain diseases and disorders. Existing statistical tools used to infer information flows are as yet far from being able to provide such insights.

This thesis provides a rigorous theoretical foundation for information flow which is designed to address the aforementioned requirements. The main contribution of this thesis is a systematic framework called *M-information flow*, which comprises a model of the brain tied to computation and a formal definition for information flow that satisfies our intuition. Through simulations of neural circuits, it is also shown how this framework can be applied in practice, and further, how we can obtain a more granular understanding of information representation by quantifying the unique, redundant and synergistic components of information about a message.

This thesis also explores theoretical and empirical connections between M-information flow and the field of causal inference. Theoretically, alternative approaches to defining information flow using counterfactual measures are established. Empirically, experiments on

artificial neural networks are used to demonstrate that the proposed measure of flow can inform interventions in simple settings. The results of these experiments indicate that the M-information flow framework can supply the necessary interpretation for diagnosing and treating brain diseases and disorders.

Lastly, this thesis considers the proposed framework in the context of existing tools used to infer information flow, such as Granger Causality. A counterexample based on communication in feedback networks is presented, wherein Granger Causality fails to infer the intuitive direction of information flow. The M-information flow framework, however, correctly recovers the expected direction of flow, while also providing deeper insight into the nature of the communication strategy. The thesis concludes with a discussion on the limitations of the proposed framework, along with potential prescriptions for overcoming some of these limitations through advances in neurotechnology.

Contents

List of Tables

List of Figures

To

My parents

1 Introduction

Philosophy is a battle against the bewitchment of our intelligence by means of language.

— Ludwig Wittgenstein

Neuroscience, over the last several decades, has undergone a technological revolution: the number of neurons we can simultaneously record has grown exponentially, mimicking Moore's law for integrated circuits [1, 2]. This trend is encouraging: it is conceivable that we will soon be in an era where we can record from the vast majority of all neurons, attain a clear understanding of their interconnectivity, and make precise targeted interventions. Indeed, much of this is already possible in some small animals [2].

This calls for thinking about how neuroscientific problems will change, and whether the theoretical and computational tools we have today are sufficient for an era with such powerful experimental capabilities. One way to imagine where neuroscience could be headed is to consider a foundational challenge in *artificial* intelligence: understanding artificial neural networks (ANNs). In an ANN, we have a system which is deterministic, free of hidden nodes, which allows for arbitrary interventions, and for which we have complete knowledge of the computations of individual units. However, even in this case, it is widely acknowledged that we do not yet truly understand the system [3, 4]. This is further borne out by the existence of the field of *explainable* (or *interpretable*) Artificial Intelligence [5]. All of this points to a need for more powerful theoretical and computational frameworks that can help us examine and understand biological and artificial neural circuits.

Within the context of this expansive problem, this thesis has a very narrow focus: developing a new framework for understanding *information flow* in neural circuits. By information flow, we intuitively mean the flow of information between the different computational units of either artificial or biological neural circuits. One of the main reasons we believe a new framework is required is that information flow has never been formally *defined* in a neuroscientific context. In fact, one might even say that the central motivation of this thesis is an examination of the simple question, *"What is information flow?"*

To explore what information flow is, and how we might define it, we will need to delve deeper into the neuroscientific context. In neuroscience, studying information flow (as understood in the intuitive sense mentioned above) is considered important because it could help us understand the brain, and eventually treat brain diseases. But in precisely what *sense* does studying information flow help us *understand* the brain?

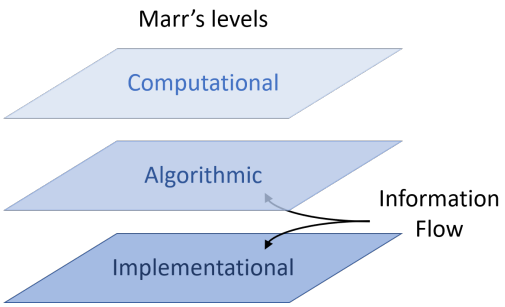


Figure 1.1: A depiction of where information flow helps support understanding, within Marr's hierarchy [6].

Herein we find that the word "understanding" is fraught with issues of its own: neuroscientists have long grappled with the question of what it even *means* to understand the brain [3, 4, 6]. Helpfully, this question has been addressed at length in the literature. Most famously, Marr and Poggio [6] proposed that the development of such an understanding takes place at three[1] distinct "levels of analysis": (i) *Computational:* recognizing the presence of a computation and decomposing it into its main components; (ii) *Algorithmic:* understanding the *algorithm* used to perform the computation within each component, including the representations that are used and how these representations are manipulated; and (iii) *Implementational:* understanding how the algorithm is implemented in a specific hardware setting, e.g., using different types of neurons and interconnections. How might a notion of information flow integrate with this framework for understanding? We posit that information flow finds involvement between the algorithmic and interventional levels (see Figure 1.1): it relates to the algorithm, since information is intrinsically tied to representation; on the other hand, it relates to the implementation because the flow is constrained by anatomy. Further, an understanding information flow can often help support or contradict different hypotheses the algorithmic and implementation levels.

As a concrete example, consider the experiment discussed by Almeida et al. [8], where the authors are trying to understand how images of common hand-held tools, such as hammers and knives, are processed by the brain (refer Figure 1.2). Like many other groups [9–14], they are analyzing how information flows in the brain while it performs a particular task. In this specific instance, they are trying to identify, albeit at a high level, which of two hypothesized algorithms the brain uses: (i) the tool can be recognized from its image alone, and information about its identity helps inform how we can manipulate it; or (ii) we can only recognize the tool by synthesizing information about its visual appearance and motor knowledge about how the tool can be manipulated. The former mechanism involves the flow of information about the identity of the tool from visual cortex to brain regions involved with object recognition, followed by the integration of motor and visual information. On the other hand, the latter mechanism involves activation of brain areas responsible for motor function *prior* to object recognition. This example makes it apparent how understanding the working of the brain, in the specific case of this this task, can be boiled down to testing between these two hypotheses, which in turn involves investigating how information about

[1] Although originally four in number, these have since been interpreted [7] more succinctly as the three levels of analysis we know today.

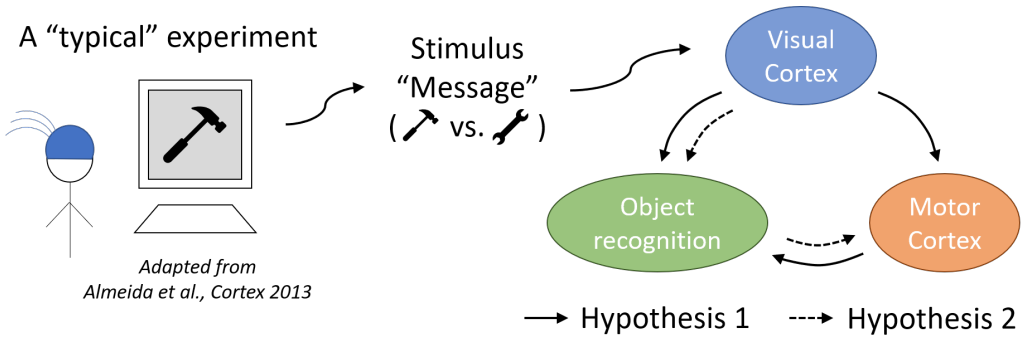


Figure 1.2: A typical neuroscientific experiment, adapted from the work of Almeida et al. [8], illustrating how investigating information flow can help us understand how the brain performs a particular task.

the tool's identity flows in the brain.

Keeping this example in mind, we can now revisit the question of how we might define information flow, and why we need a framework for it. The aforementioned experiment gives us a few anchors to help determine what constitutes a good definition of information flow, and how we should model a neural circuit:

1. Firstly, the experiment suggests that our neural circuit models could consist of distinct “brain regions” that perform computations and send information between each other. In the example above, the regions refer to large well-separated regions like visual and motor cortex, but in other experiments [15], we may be interested in flows between smaller groups of neurons at a finer scale. In our work, we model the neural circuit as a computational graph that is able to capture flows at various levels of abstraction.

2. Secondly, we are interested in the specific *paths* along which information flows in the circuit. Therefore, any definition of information flow we provide must be able to consistently track information along paths. This requirement becomes a defining theme of our work: one that is in fact harder to satisfy than we might intuitively expect.

3. Thirdly, we are often trying to understand how information about some specific *message* flows in the brain, for example, a stimulus or a behavioral response. In the example above, the message was that information which distinguished different stimuli, namely, the identity of the tool. This idea, that the information flow is usually *about* one or more specific *messages*, plays a central role in our setup and definitions, and is a distinguishing feature of our work relative to methods such as Granger Causality.

4. Another important aspect illustrated by the aforementioned example is the temporal nature of information flow: the flow can change from instant to instant, and we are interested in tracking these temporal differences. A related issue has to do with the possibility of feedback in the neural circuit, which a definition of information flow should be able to account for.

5. Furthermore, when dealing with a richer stimulus (e.g., visual stimuli with different colours and shapes, or natural scenes), one might wish to understand the information

flow of each of the stimulus' components. Alternatively, if there are two paths that appear to carry information about the stimulus, one might wish to examine what information is unique to each path, and what information is redundantly contained in both paths.

The example provided above shows how information flow can be useful in neuroscience, for understanding the brain at the interface of Marr's algorithmic and implementational levels. However, a nuanced understanding of information flow in the brain could also help with diagnosing and treating brain diseases [13, 14, 16, 17]. For instance, such an understanding may be essential when considering interventional approaches to treating dysfunctional components of the nervous system. These interventions could take many forms: conventional drugs, electric or magnetic stimulation, or neurofeedback techniques that make use of repeated stimulus presentation [18, 19] are but a few. When considering electrical stimulation specifically, prime examples of devices already in use include responsive neurostimulation for epilepsy [20] and deep-brain stimulation for Parkinson's disease [21]. Similar stimulation-based techniques are also starting to gain momentum for treating disorders such as addiction and depression [21, 22]: in the wake of the opioid crisis in the United States [23], understanding information flow in the brain's reward networks could be critical to finding the right location and signal parameters for electrical stimulation [24, 25].[2]

Therefore, we believe that information flow is a useful subject of study, and indeed, it finds common usage in neuroscientific parlance. In the literature, popular approaches for inferring information flow include using measures of statistical causal influence such as Granger Causality [26, 27], Transfer Entropy [28] or Directed Information [29–32]. But for a number of reasons, which we detail in Section 2.1.3 and examine more closely in Chapter 6, these measures cannot be interpreted as information flow, at least not in the sense motivated above. Because they do not satisfy the aforementioned requirements, these measures fail to provide the degree of insight and interpretability we seek. Despite this, there has not been a concerted effort to formally define information flow in a neuroscientific context.

The lack of such formal terminology has been noted before, as an impediment to progress in biology, by Lazebnik [33] in an influential paper titled "Can a biologist fix a radio?". Lazebnik argues that, in engineering fields, "formal language unites the efforts of many individuals" and that the standardization of language allows for communication that is unambiguous. In its absence, statements are necessarily vague and the ability to make clear predictions is impeded. This thesis hopes to introduce such universal terminology for information flow in neuroscience, by developing a computational system model and a mathematically rigorous definition through careful introspection. The process of developing such a framework requires that we clearly state the assumptions of the model, which might otherwise be implicit in informal usage of terms like "information flow". We believe that this enables us to draw much more concrete interpretations from information flow inferences.

[2] Several aspects of the motivation for understanding information flow in this thesis can be attributed to Prof. Rob Kass, who explains the need for new frameworks to address this question in his 2017 COPSS Fisher Lecture (https://youtu.be/_2EyHnua0W4).

1.1 An Outline of this Thesis

The thesis begins with Chapter 2, which provides a framework for defining information flow in such a way as to satisfy the intuitive requirements mentioned above. This definition will capture the information flow about a specific *message*, e.g., the stimulus, within a computational system designed to model the brain. Using a series of candidate definitions and counterexamples, we show that an important aspect of defining information flow in a meaningful way is to recognize and account for *synergistic* information representation, wherein information is communicated using two parallel edges jointly, and cannot be detected on either edge individually. By accounting for synergy using a definition based on conditional mutual information, we prove that it is always possible to track the information flow about a given message. We also provide a number of canonical computational examples to show how this definition of information flow satisfies our intuition.

Chapter 3 presents simulations to show that our definition of information flow can be applied in a neuroscientific context. These simulations use networks of neurons, similar to quadratic-integrate-and-fire neurons, to illustrate how synergy might arise in neural circuits and how our framework can help identify information flow in such settings. Furthermore, we address how partial information measures, i.e., unique, redundant and synergistic information, can help provide fine-grained insights about information representation that cannot be obtained using existing statistical or information-theoretic tools. These results also rely on simulations, but of spatial encoding in entorhinal grid cells, and reveal that all three forms of partial information may arise in these cells depending upon how they encode information.

Chapter 4 serves to demonstrate that our information flow framework can be easily adapted to measure the flows of information in artificial neural networks. Leveraging the context of fairness in artificial intelligence, we show in this segment that although our measure of information flow is observational, it correlates with the outcome seen upon pruning edges in simple networks. This suggests an operational interpretation for information flow in simple artificial neural networks, namely, that edges with larger information flow about a particular message are more likely to be responsible for how that message influences the output. In particular, this result could have implications for using the knowledge of flows to inform interventions for diagnosing and treating brain diseases and disorders.

Chapter 5 explores alternative definitions to information flow and its theoretical connections with the field of Causality [34, 35]. Specifically, we examine how we might overcome a drawback of the information flow definition, i.e., the existence of nodes at which flow is not conserved (called *information orphans*), and identify alternative definitions that do not suffer from this issue. We show that even common-sense approaches based on pruning fail to work due to the presence of non-intuitive counterexamples. In particular, counterfactual causal influence proves to be a powerful way to understand information flow, which satisfies all our intuitions and does not suffer from "orphans", although such a definition cannot be estimated in practice. We also provide examples to show the relationship (or lack thereof) between our definition of information flow and counterfactual causal influence.

Finally, in Chapter 6, we examine Granger Causality in greater detail, providing a counterexample wherein the direction of information flow is opposite to the direction of greater (or even statistically significant) Granger causal influence. This counterexample is based on a well-known feedback communication scheme known as the Schalkwijk-Kailath

strategy. We also show how the information flow framework resolves this counterexample, obtaining not only the intuitively "correct" directions of information flow, but also revealing insights about the underlying computation.

In Chapter 7, we discuss of the limitations of the proposed framework, and offer potential means to mitigate these limitations through the development of new neurotechnologies.

1.2 Related Works of the Author

In an effort to keep this thesis contained, several works of the author that were only tangentially related to information flow were omitted. What follows is a brief summary of these works and an explanation of the common thread running through all of them.

A significant fraction of the author's Ph.D. years were spent working on understanding and improving neurotechnologies, specifically, high-density electroencephalography (EEG). This research addresses a long-standing question: what is the best possible "resolution" we can achieve in EEG source localization, and how does this resolution *scale* with EEG sensor density? Prior attempts to answer this question have relied on a spatial Nyquist-rate analyses [36]; however, these analyses failed to account for different types of noise, and gave reduced importance to high spatial frequencies. We believe that this has contributed to a pervasive view that increasing EEG sensor density does not help improve spatial resolution.

We have sought to change this perception through a number of studies. First, using simplified brain models, we found that as long as measurement noise was sufficiently low, one can obtain arbitrarily high resolutions with increasing sensor density [37]. This is contrary to conventional wisdom in the field, which suggests that spatial information is irretrievably blurred by volume conduction through the skull. Instead, our results show that information relevant to high-resolution reconstructions is never really *lost*, rather it is simply obscured by noise, which calls for controlling measurement noise in EEG. Our analysis also revealed the importance of estimating the power spectral density of spatial noise in EEG, which directly affects our ability to resolve the spatially blurred signal, as well as the parameters of our algorithm [37].

While the previous work indicates that information is to be gained by going to higher densities, it is unclear how far we can push these limits: what is the best possible resolution that can be obtained for *source localization*, with a fixed number of EEG sensors? Using an information-theoretic approach, we derived the first *minimax lower bounds* on source localization error that scaled with the number of sensors—such bounds establish what resolutions are *unattainable* at a given sensor density [38]. Complementarily, deriving these bounds can inform the design of new source localization *algorithms* that can achieve resolutions close to the bound. These results supported our earlier work [37], and have been subsequently improved upon by other groups [39].

We have also shown through experiments that high-density EEG can provide more information than low-density EEG in neuroscientific experiments [40]. This experiment collected EEG data from participants fitted with a specially reconfigured high-density EEG cap, wherein all 128 electrodes had been reconfigured to lie over occipital and temporo-occipital areas. Participants were shown counterphased circular checkerboard patterns of different spatial frequencies, and asked to perform an orthogonal task to ensure attention. Through a series of classification-based analyses, we established that the high-density EEG

configuration was able to distinguish between the stimuli of different spatial frequencies better than four low-density EEG configurations, which were subsampled from the original 128 electrodes.

These theoretical and experimental studies gave rise to several new lines of interdisciplinary research: close collaborations with circuits and systems engineers have yielded new subdermal EEG systems [41] and easy-to-install high-density EEG caps [42]. Both of these works were directly inspired by the broader theoretical promise of high-density EEG, as well as our efforts to find efficient implementation strategies [43]. The possibility of being able to localize slowly-varying signals non-invasively also motivated an effort to non-invasively detect cortical spreading depolarizations using an automated algorithm, which could help identify and mitigate worsening brain injury [44].

Having discussed how we might develop better neurotechnology for sensing brain signals, we now proceed to analyze improved methods for making interventions, once again in the clinical domain. Responsive Neurostimulation (RNS) is an effective treatment for drug-resistant epilepsy patients who are not suitable candidates for resection surgery. However, a lack of objective measures of its effectiveness has hampered our ability to tune stimulation parameters to provide patients a quicker recovery, while also providing us a better understanding of how RNS helps control seizures. To address this issue, we developed a metric based on Wasserstein distances [25] to quantify *indirect frequency modulation*, a recently established biomarker of patient responsiveness to RNS [45]. Our algorithm identified clear patterns of increased Wasserstein's distances in patients expert-identified to show indirect frequency modulation, indicating the potential of our method to predict long-term patient outcomes. Such a method could be used in future as part of a closed-loop system, to get objective feedback on the course of recovery for each patient, allowing us to optimize the RNS settings and deliver more personalized treatment.

The theme of using information-theoretic methods to develop new measures and frameworks also extends to some of our other research directions. In the context of fairness in artificial intelligence, our work has sought to define new measures of how biased machine learning algorithms are against protected groups [46, 47]. These works focus on quantifying bias, while allowing certain *critical* features to be exempt from contributing to such bias: these could be features that must be included for safety reasons, for example, even if they currently contribute to unfairness against a protected class. This quantification relies on a combination of tools from the literature on partial information decomposition, such as uniqueness and synergy, as well as Causal inference. This work is not only close in spirit and methodology to much of our work on information flow, but as shown in Chapter 4, information flow could direct application in identifying paths along which bias propagates, and could offer strategies for incorporating fairness in such frameworks.

Finally, the author has begun to contribute to the understanding and development of partial information measures [48]. As seen in various parts of this thesis (refer Chapters 2 and 3), as well as in the aforementioned work on quantifying bias in AI, the development of interpretable partial information measures holds great potential for widespread application.

2 The M-Information Flow Framework

All models are wrong, but some are useful.

— George Box[1]

2.1 Introduction

This chapter forms the central pillar of this thesis, laying out a new framework for understanding information flow in neural circuits.[2] To let the chapter stand as an independent unit, we restate and elaborate upon the motivations presented in Chapter 1. The introductory sections below also help differentiate our framework from existing measures used to interpret information flow, such as Granger Causality. A complete outline of the chapter is provided in Section 2.1.4.

2.1.1 Motivation

Neuroscientists often seek an understanding of how information flows in the brain while it performs a particular task [8–12]. As a concrete example, consider the experiment performed by Almeida et al. [8], where they examine how images of common handheld tools are processed in the brain. In simple terms, the question they investigate is this: when attempting to identify a handheld tool, does one make use of knowledge of how to manipulate it? Two hypotheses present themselves: (i) the answer to the above question is *yes*, so we should expect that information about a tool's identity *first* flows from visual cortex to motor cortex (the area responsible for processing manipulation), *before* synthesis of visual and motor information occurs at the area of the brain responsible for object recognition; (ii) alternatively the answer to the aforementioned question is *no*, so we should expect that the information about tools' identities *first* flows from visual cortex to the area responsible for object recognition, *after* which this information arrives at motor cortex. Thus, distinguishing between these hypotheses is equivalent to *determining the path* along which information about a tool's identity flows in the brain. What methods can neuroscientists use to gain

[1] Although George Box is usually given credit for it, forms of this saying have predated his writings.

[2] This chapter is based largely on our paper published in the *IEEE Transactions on Information Theory*, titled "Information Flow in Computational Systems" [49].

such an understanding? What formal theory underlies such an analysis? How does one mathematically define colloquially-used terms such as "information flow"? These are the fundamental questions we try to answer in this paper.

As another example, consider the work of Hong et al. [9], who show that mice can detect the presence of an object with their whiskers, even without their "barrel cortex", the primary sensory area of the brain responsible for this task. It is hypothesized that information about the object's presence passes from neurons at the whiskers through an alternative pathway involving deeper brain regions that add redundancy to the system. How is sensory information about the presence of external objects encoded between the cortex and these deeper regions? How much of the information flow is expressed uniquely in each area, and how much of it is redundant? And once again, how do we systematically discuss the measurement of information flow along each of these pathways?

Information flow is a concept that appears in several contexts, across fields ranging from communication systems [50, 51], control theory [52, 53] and neuroscience [8–12] to security [54], algorithmic transparency [46, 55], and deep learning [56–58]. While our primary motivation comes from neuroscience, the theory that we develop is broadly applicable to any system which can be modeled in the form of a directed graph, with nodes that communicate functions of their inputs to other nodes, and where transmissions are observable. For example, several kinds of social networks readily fit this bill, and one might wish to analyze the spread of (mis)information or infectious disease in such networks [59, Ch. 16]. Our framework is also general enough to analyze information flow in various kinds of Artificial Neural Networks: this could be useful for identifying specific paths that carry information distinguishing two or more classes, or for intelligently pruning an Artificial Neural Network post-training [60, 61].

In the field of neuroscience, studying the paths along which information flows could be essential for understanding, diagnosing and treating brain diseases [13, 14, 16, 17]. For example, understanding information flow pathways is essential when considering principled approaches for intervening to affect the output of a computational system, as is done in Responsive Neurostimulation in Epilepsy [20] or Deep Brain Stimulation in Parkinson's Disease [62], or for complementing dysfunctional aspects of the nervous system, such as in cochlear [63] and retinal implants [64].

2.1.2 Our Goal and Approach

Our overarching goal in this paper is develop a formal theory for understanding information flow in neuroscientific experiments. Based on the examples highlighted in the previous section, we can summarize the key aspects of what we mean by "information flow". We want to capture:

1. information flow between *distinct nodes*, representing different brain regions at some scale;

2. information about a *specific message*, i.e., the stimulus;

3. specific *paths* along which information flows;

4. *dynamics*, i.e., there may be information flow at one time instant, but not at a later instant; and

5. *feedback*, i.e., information about a message may flow back and forth between two areas.

In what follows, we expand upon some of these points, and outline our methodology for designing a computational model and providing a definition of information flow that addresses these issues.

In order to properly scope our task, we choose to restrict our attention to "*event related experimental paradigms*" [65]. These are a set of standard neuroscientific experimental protocols where a timed *stimulus* is presented to an animal subject or human participant over multiple trials, while their brain signals are being recorded. Restricting ourselves to such experiments allows us to decide precisely what *kind* of information flow we are interested in, since in general, the phrase "information flow" can refer to more than one notion in neuroscience. We identify two dominant interpretations of "information flow": (i) the first refers to information about a *specific* quantity or variable that is of interest to the experimentalist, which in this paper we refer to as the *"message"*; (ii) the second refers to information in the *abstract*, and is usually used to describe the fact that one area of the brain "drives" or "influences" another area through the transmission of some information: in this interpretation, one is not interested in *what* is being communicated, only that the communication is *occurring*. In this paper, we focus only on the first interpretation of the phrase, where we are interested in information about a *specific message*. This is particularly common in event-driven paradigms, where the neuroscientist investigates how the brain responds to a carefully chosen set of stimuli, and examines the paths along which information contained in these stimuli[3] flows through the brain.

We approach our goal of providing a theoretical framework for information flow by formally defining a computational system model. This model is based on a graph consisting of nodes, representing distinct computational areas of the brain, and edges, representing the connections between them. The nodes of this graph can potentially represent the brain at any scale: single neurons, groups of neurons, or even whole brain regions, depending on the measurement modality and the kind of experiment being performed. These nodes compute stochastic functions of transmissions received from their incoming edges, and send the results of these computations on their outgoing edges. The idea of the computational graph is inspired from Thompson's work on VLSI complexity theory [66], while the model for stochastic computation is derived from Structural Causal Models in the field of Causality [34, 35].

Next, we find a formal definition for information flow that satisfies the requirements listed above. In order to attain a *dynamic* picture of information flow on the edges of the computational graph, and to deal with *feedback* in the flow of information, we use the idea of time-unrolling the computational graph (taking inspiration from Network Information Theory [50]). Principal among our requirements is that we must be able to *track, using an unbroken path, how information about the message flows through the system.* Imposing this requirement as a desired property, we iterate through a series of candidate definitions and counterexamples, finally arriving at a definition based on conditional mutual information, which satisfies the aforementioned requirements.

[3] or alternatively, information contained in the *response*

Given that we are interested in *information about a message*, we rely on information-theoretic measures to define information flow. Furthermore, we restrict ourselves to "*observational*" measures, which can be computed from a sample of all random variables described in the model. We deliberately eschew *interventional* and *counterfactual* measures: the former require the capability to intervene on the system and change the distributions of the random variables involved; meanwhile the latter are usually theoretical, and can only be applied in situations where one can ask what *might have occurred* if a specific variable had been different on a particular trial (while keeping the realizations of all unobserved private sources of randomness fixed).

The approach of building a rigorous theoretical framework that we have adopted in this paper is inspired by two works from biologists titled "Can a biologist fix a radio?" [33] and "Could a neuroscientist understand a microprocessor?" [67]. Both these works point to the lack of formal methods, i.e., systematic theory, that could help biologists understand the limitations of their tools and test their assumptions. It is our belief that information theory can help provide the formal methods that are sought in biology, and make an impact in fields such as neuroscience and neuroengineering [37, 38, 43]. In particular, information theory can play an important role in advancing how we understand large computational systems through external measurements and interventions. While developing an understanding of information flow in such systems may not be sufficient for providing a complete description of the nature of computation itself, we believe that it forms an integral component. Providing a formal theoretical framework for information flow is but a small part of several broader theoretical questions that are yet to be properly posed: questions such as how one might formalize "reverse engineering" the brain, or formalize the notion of "understanding computation".

2.1.3 Related Work

Prior work on statistically inferring flows of information in the brain appears under the umbrella of "functional" or "effective connectivity" [68–70]. These efforts have largely relied on measures of statistical[4] causal influence such as Granger Causality [26, 27], Massey's Directed Information [29–32], Transfer Entropy [28] and Partial Directed Coherence [71]. Despite widespread use, these measures have frequently been a subject of debate and disagreement within the neuroscientific community [72–78]. In part, these disagreements stem from the widely-acknowledged fact that under non-ideal measurement conditions (e.g. in the presence of hidden variables [34, p. 54], asymmetric noise [79, 80], or limited sampling [81]), estimation of these quantities may be erroneous. While these non-idealities may eventually be overcome through improvements in technology, we believe that more fundamental issues still remain. For instance, one basic question that has remained unanswered is: when can statistical causal influence be interpreted as information flow about a message? In previous work, we demonstrated that even under *ideal* measurement conditions, the direction of greater Granger causal influence can be opposite to the direction in which the message is being communicated in certain kinds of feedback communication networks [82]. This example points to a more general issue with the use of statistical causal influence measures: there is no direct way to interpret what the influence is *"about"*. While it is understood in

[4] We borrow the use of the term "statistical" from Pearl [34, Sec. 1.5], who contrasts and differentiates "statistical" concepts from (strictly) "causal" ones.

certain settings that "information flow" refers to information contained in a particular set of *"stimuli"* (as mentioned in the previous section), the aforementioned measures do not incorporate the effect of the stimulus.

The existence of such fundamental issues can be traced back to the fact that there is *no underlying model* that links information flow (of some message of interest) with the signals that are actually *measured*. This leads to a lack of separation between the problems of *defining* information flow and of *estimating* it, while also making it hard to test assumptions and draw the right interpretations from experimental analyses. We believe that, just as Shannon provided a theoretical foundation for information transmission [83], a solid theoretical treatment of information flow is needed. Such a treatment would begin with a model of the underlying system, give a definition for information flow and describe its properties, and finally end with a suitable estimator. Adopting Shannon's approach of defining entropy by stating a set of properties that such a measure must satisfy, we attempt to define information flow by putting forward an intuitive property that we believe is desirable for such a quantity. It is our hope that, by providing a theoretical foundation that separates definition and estimation, along with a concrete model and explicitly-stated assumptions, we can avoid many of the pitfalls encountered by previous approaches to understanding information flow in the brain.

It is useful at this point to mention the key differences between our measure of information flow, and measures based on Granger Causality and its generalizations:

1. Our measure depends on a message, M, that is related to the stimulus or the response in a neuroscientific task, whereas tools based on Granger Causality do not.

2. Since Granger Causality-based tools use time series modeling to compute an estimate of information flow, they are unable to provide a dynamic, evolving picture of information flow between different areas over time (although we must acknowledge that there have been recent efforts towards bridging this gap [84]).

3. Since we start with a *computational framework*, our model provides a direct way to connect information flow with the underlying computation. On the other hand, Granger Causality-based tools start with a probabilistic graphical model of the observed nodes, and do not tie the analysis to computation in any way.

While our proposed definition of information flow will also suffer from performance degradation under non-ideal measurement conditions, we believe that it overcomes the fundamental difficulty faced by Granger Causality-based tools: when measurements are ideal, our definition provides a clear and consistent way to interpret information flow about a message, as we illustrate through several examples in Section 2.6.

Another line of work that appears within the functional and effective connectivity literature is Dynamic Causal Modeling (DCM) [69, 85]. This methodology is, in spirit, much more closely aligned with what we propose here. However, our framework differs from DCM in a few important ways: (i) our underlying framework and model is based on Structural Causal Models rather than dynamical systems, and (ii) we seek to formalize the notion of information flow, not just of effective connectivity. However, the style of thinking, which involves starting from theoretical models and incorporating the stimulus and experimental design, is common to both DCM and our approach.

2.1.4 Outline of the Paper

In this paper, we start by giving a mathematical description of a generic computational system, about which inferences are being drawn (Section 2.2). We then formally define what it means for information about some message to flow on a single edge or on a set of edges in the computational system (Section 2.3). This is done by proposing an intuitive property that we would like such flows to satisfy, along with some candidate definitions, and then examining which candidates satisfy the property. The intuitive property we desire is: *information flow about a message may not completely disappear from the system at a certain time, only to spontaneously reappear at a later point* (formalized in Property 2.1). It emerges that simple and intuitive definitions actually fail to satisfy this basic property, and so a more sophisticated definition is needed. We then show how our definition for information flow about the message satisfies several desirable properties, including guarantees for the existence of "information paths" between appropriately defined input and output nodes (Section 2.4). We also show how our definition has a very non-intuitive property—information about a message may flow *out* of a node despite not flowing *into* it—and justify why this might be reasonable for an observational definition. After that, we suggest how one might detect which edges of the computational system have information flow, and provide an "information path algorithm", which identifies the aforementioned information paths (Section 2.5). We also introduce and discuss the concepts of derived information, redundant transmissions and hidden nodes, which allow one to obtain a more fine-grained understanding of information structure in the computational system. To show that our definition of information flow agrees with intuition, we give several canonical examples of computational systems and depict the information flow in each case (Section 2.6). Finally, we conclude with discussions on connections with neuroscience, issues related to the difficulty of estimating information flow (along with possible remedies), comparisons with the existing statistical causal influence literature, connections with fields such as probabilistic graphical models and causality, and a discussion on information volume (Section 2.7).

2.2 The Computational System

Our goal is to develop a rigorous framework for understanding how information about a message flows in a computational system. To do this, we first need to define the terms "computational system", "message", "information about a message" and "flow". In this section, we start with the first two terms, defining the model of the computational system that is used throughout this paper, and explicitly defining the message.

Our model is based on prior art in the information theory literature [50, 66], and consists of nodes communicating to each other at discrete points in time on a directed graph. At every time instant, each node receives transmissions on its incoming edges and computes a function of these transmissions to send out on its outgoing edges. This function can be random and time-dependent, and can be different for every outgoing edge. We will be interested in the flow of a particular random variable called the "message", which will be defined shortly. Since the directed graph forming the computational system may have cycles, the message may flow along a cyclic path. To deal with this possibility while capturing the

fact that nodes must be causal,[5] we define a "time-unrolled" graph (in a manner similar to Ahlswede et al.[6] [50]), which describes how nodes communicate to each other over time. We define a random variable model for the nodes' transmissions, and demonstrate how each node computes these variables. We also formally define the input nodes of the computational system, through their relationship with the message.

Definition 2.1 (Complete directed graph)**.** *A* complete directed graph $\mathcal{G}^* = (\mathcal{V}^*, \mathcal{E}^*)$ *is described by a set of nodes and the set of all edges between those nodes (including self-edges). We denote the set of nodes by their indices,* $\mathcal{V}^* = \{1, 2, \ldots, N\}$*, where* N *is a positive integer denoting the number of nodes in the graph. The set of edges in the graph is the set of all ordered pairs of nodes,* $\mathcal{E}^* = \mathcal{V}^* \times \mathcal{V}^*$*.*

Note that (i) edges are directed, so the edge $(A, B) \in \mathcal{E}^*$ describes an edge *from* node A *to* node B; and (ii) nodes have self-edges. For every $A \in \mathcal{V}^*$, there is an edge (A, A) in $\mathcal{E}^*$.

Moving forward, nodes shall be thought of as performing computations and possessing local memories. We shall interpret the transmission of a node to itself as the variable it stores within its memory.[7]

Definition 2.2 (Time-unrolled graph)**.** *In order to allow nodes to have different transmissions at every time instant, we must provide for the progression of time. Let* $\mathcal{T} = \{0, 1, \ldots, T\}$ *be a set of time indices, where* T *is a positive integer representing the maximum time index. Then, a* time-unrolled graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ *is constructed by indexing a complete directed graph* $\mathcal{G}^*$ *using the time indices* $\mathcal{T}$ *as follows:*

1. *The nodes* $\mathcal{V}$ *consist of all nodes* $\mathcal{V}^*$ *in* $\mathcal{G}^*$*, subscripted by time indices in* $\mathcal{T}$*,*

$$\mathcal{V} = \{A_t : A \in \mathcal{V}^*, t \in \mathcal{T}\};$$

2. *The edges* $\mathcal{E}$ *connect nodes of* successive *times in* $\mathcal{V}$*, so they can be written in terms of the edges in* $\mathcal{E}^*$ *as*

$$\mathcal{E} = \{(A_t, B_{t+1}) : (A, B) \in \mathcal{E}^*, t \in \mathcal{T} \setminus \{T\}\}.$$

For brevity, we denote the set of all nodes at time t by $\mathcal{V}_t$, and the set of all (outgoing) edges at time t by $\mathcal{E}_t$. So, for example, we will have $A_1 \in \mathcal{V}_1$ and $(A_1, B_2) \in \mathcal{E}_1$. All of the notation in this section can be visualized in Figure 2.1 and is summarized in Table 2.1.

Once again, note that (i) edges at time t connect nodes at time t to nodes at time $t + 1$; and (ii) since the original graph $\mathcal{G}^*$ had self-edges, there will always be an edge (A_t, A_{t+1}) in $\mathcal{E}_t$ for every node $A_t \in \mathcal{V}_t$.

[5] Causal in the "Signals and Systems" sense of the word, where a node cannot make use of future transmissions [86].

[6] Although the work of Ahlswede et al. (2000) is titled "Network *Information Flow*", it actually addresses a different problem: one of the achievable rate region of a broadcast network and the optimal coding strategy that achieves this rate. In contrast to their work, which concentrates on characterizing and achieving the optimal rate, our focus is on understanding how information about a known message flows in an existing computational system.

[7] Instances of directed graphs that are *not complete* and of nodes possessing *no memory* are merely special cases of our model, where the respective edges' transmissions can simply be set to zero.

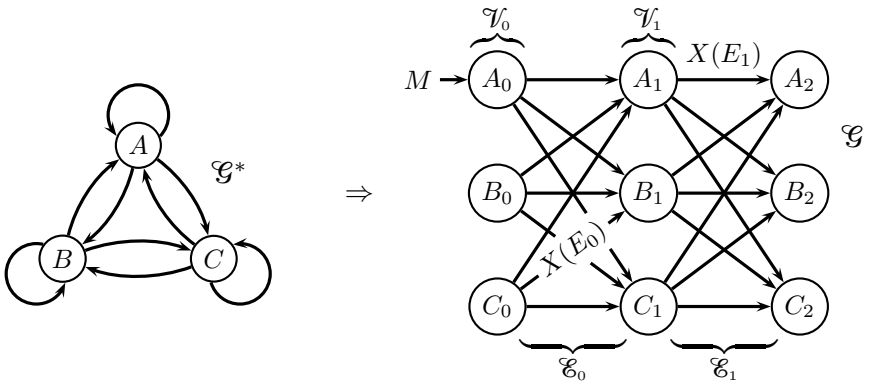


Figure 2.1: A diagram showing an example of a how a complete directed graph is unrolled to create a time-unrolled graph. On the left, we show a complete directed graph $\mathcal{G}^*$ that has three nodes, $\mathcal{V}^* = \{A, B, C\}$. These nodes are fully connected to each other via edges $\mathcal{E}^*$, including self-edges.
On the right, we show how $\mathcal{G}^*$ has been unrolled using time indices $\mathcal{T} = \{0, 1, 2\}$ to obtain a time-unrolled graph $\mathcal{G}$. The set of all nodes at time $t = 0$ is $\mathcal{V}_0$ and the set of all (outgoing) edges at time $t = 0$ is denoted $\mathcal{E}_0$. As an example, we have shown an arbitrary edge $E_0 \in \mathcal{E}_0$ (here, $E_0 = (C_0, B_1)$) and the transmission on that edge, $X(E_0)$. As another example, we show a "self-edge" in the time-unrolled graph, $E_1 \in \mathcal{E}_1$, which in this case is $E_1 = (A_1, A_2)$. Also depicted is the transmission $X(E_1)$ on this self-edge, which is interpreted as the contents of the memory of node A from $t = 1$ to $t = 2$. The message M arrives at the input node A_0, but could in general be available at more than one node at $t = 0$.
In subsequent illustrations, we do not depict all edges at every time step, even though they are present. This is done only for the sake of clarity.

Also note, we have only presented the complete directed graph in Definition 2.1 in order to explicitly define the process of time-unrolling. We do not expect the time-unrolled graph to be "rolled back" into a complete directed graph at the end of an information flow analysis. Since we seek a time-evolving picture of information flow between different computational nodes, we will directly view and interpret information flow on the time-unrolled graph. This is illustrated later, through several examples, in Section 2.6.

Definition 2.3 (Computational System)**.** *A* computational system $\mathcal{C} = (\mathcal{G}, X, W, f)$ *is a time-unrolled graph* $\mathcal{G}$ *that has* transmissions on its edges *which are constrained by* computations at its nodes*. The* input *to the computational system includes a* message,[8] M*. We now elaborate upon these terms:*

2.3a) Transmissions on Edges

We begin by defining a function which maps every edge of $\mathcal{G}$ *to a random variable. Let* $\mathfrak{X}$ *be the set of all random variables in some probability space.*[9] *Then, let* $X : \mathcal{E} \to \mathfrak{X}$ *be a function that describes what random variable is being transmitted on a given edge, i.e.,* $X(E)$ *is the random variable corresponding to the transmission on the edge* E*.*

For convenience, we define X *applied to a* set of edges *as the set of random variables produced by applying* X *to each of those edges individually, i.e., for any set* $\mathcal{E}' \subseteq \mathcal{E}$*,*

$$X(\mathcal{E}') = \{X(E) : E \in \mathcal{E}'\}. \tag{2.1}$$

[8] The message is the random variable *whose* "information flow" we will seek to identify.

[9] We assume that all probability distributions are such that the mutual information and conditional mutual information between any sets of random variables is well-defined [87, Sec. 2.6].

We extend the use of this notation to other functions of nodes and edges that we define, going forward.

2.3b) Computation at a Node

Let $A_t \in \mathcal{V}_t$ be a node in the time-unrolled graph $\mathcal{G}$, at some time $t \geq 1$ (recall that $t \in \{0, 1, \ldots, T\}$). Let $\mathcal{P}(A_t)$ be the set of edges entering A_t, and $\mathbb{Q}(A_t)$ be the set of edges leaving A_t. Further, let us suppose that A_t is able to intrinsically generate the random variable[10] *$W(A_t)$ at time t, where $W(A_t) \perp\!\!\!\perp W(\mathcal{V} \setminus \{A_t\})\ \forall\ A_t \in \mathcal{V}$, $W(\mathcal{V}_t) \perp\!\!\!\perp \{M\} \cup \{X(\mathcal{E}_{t'}) : t' \in \mathcal{T}, t' < t\}$ and the symbol "$\perp\!\!\!\perp$" stands for independence between random variables. Then, the* computation *performed by the node A_t (for $t \geq 1$) is a deterministic function*[11] *f_{A_t} that satisfies*

$$f_{A_t}(X(\mathcal{P}(A_t)), W(A_t)) = X(\mathbb{Q}(A_t)). \tag{2.2}$$

Here, $X(\mathcal{E}_{t-1})$, $W(\mathcal{V} \setminus \{A_t\})$, $W(\mathcal{V}_t)$, $X(\mathcal{P}(A_t))$ and $X(\mathbb{Q}(A_t))$ all make use of the notation described in (2.1).

Note that the definition above does not apply when $t = 0$; this is a special case which is discussed below. Also, for convenience, where $\mathcal{A}$ is an arbitrary set of nodes, we will use $f_{\mathcal{A}}$ to denote the "joint function" mapping the incoming transmissions of all nodes in $\mathcal{A}$ (along with their intrinsic random variables $W(\mathcal{A})$) to their respective outgoing transmissions.

2.3c) The Message and the Input Nodes

The message *is a random variable M, which is of interest to the experimentalist observing the computational system, and for which we shall define information flow. For now, we assume that we are interested in a single message.*[12] *We also assume that the message enters the computational system only at time $t = 0$, and at no later time instant.*

We formally define the input nodes *of the system as those nodes of $\mathcal{G}$, at time $t = 0$, whose transmissions statistically depend on the message M:*

$$\mathcal{V}_{ip} := \{A_0 \in \mathcal{V}_0 : I(M; X(\mathbb{Q}(A_0))) > 0\}, \tag{2.3}$$

where $\mathbb{Q}(A_0)$ represents the set of edges leaving the node A_0.

To remain consistent with Definition 2.3b, we define the computation performed by an input node $A_0 \in \mathcal{V}_{ip}$ as a function f_{A_0} that satisfies

$$f_{A_0}(M, W(A_0)) = X(\mathbb{Q}(A_0)), \tag{2.4}$$

[10] $X(E_t)$ and $W(A_t)$ may also be random *vectors* instead of random variables, i.e., an edge may *transmit a vector*. This does not affect the theoretical development presented here; all of our proofs remain unchanged.

[11] This kind of model is not new. For instance, in the causality literature, it is known by a few different names: Pearl refers to it as a "Structural Equation Model" [34, Sec. 1.4.1], while Peters et al. refer to it as a "Structural Causal Model" [35]. We prefer the latter terminology, which makes explicit the connection to causality.

[12] That is, we assume that the message is a single random variable or vector. It is possible to simultaneously examine the information flows of several (possibly dependent) messages, or of sub-messages within a single message. These cases are examined in Section 2.5.6.

Table 2.1: Summary of Notation

Variable(s)	Meaning
$\mathcal{G}^* = (\mathcal{V}^*, \mathcal{E}^*)$	The original complete directed graph, prior to time-unrolling
$\mathcal{G} = (\mathcal{V}, \mathcal{E})$	The time-unrolled graph making up the computational system
$\mathcal{T}$	The set of all time points, $\{0, 1, \ldots, T\}$
$\mathcal{V}$	The set of all nodes in the computational system
$\mathcal{V}_t$	The subset of nodes at time t
V_t, A_t, B_t, C_t, D_t	A node in the graph at time t
V, A, B, C, D, E	A node in the original complete directed graph $\mathcal{G}^*$, or a node in the computational system at an unspecified time point
$\mathcal{A}, \mathcal{B}$	Some subset of nodes in $\mathcal{V}$
$\mathcal{E}$	The set of all edges in the computational system
$\mathcal{E}_t$	The set[†] of all edges at time t
$\mathcal{E}'_t$	Some subset[‡] of edges in $\mathcal{E}_t$
E_t, P_t, Q_t, R_t, S_t	An edge in the computational system at time t
E, P, Q, R, S	An edge in the original complete directed graph $\mathcal{G}^*$, or an edge in the computational system at an unspecified time point
$X(E_t)$	The random variable representing the transmission on the edge E_t
$X(\{E^{(1)}, E^{(2)}\})$	Short-hand notation for $\{X(E^{(1)}), X(E^{(2)})\}$ (refer Equation (2.1))
$\mathcal{P}(V_t)$	The set of all incoming edges of V_t $(= \mathcal{V}_{t-1} \times \{V_t\} \subseteq \mathcal{E}_{t-1})$
$\mathcal{Q}(V_t)$	The set of all outgoing edges of V_t $(= \{V_t\} \times \mathcal{V}_{t+1} \subseteq \mathcal{E}_t)$
$W(V_t)$	The intrinsically generated random variable at the node V_t
M	The "message", a random variable that enters the system at time $t = 0$, and whose information flow we seek to understand (refer Definition 2.3c)
$\mathcal{V}_{\text{ip}}$	The input nodes: the subset of nodes at time 0 whose outgoing transmissions depend on the message M (refer Definition 2.3c)
f_{V_t}	The function computed by the node V_t (refer Definition 2.3b)

[†]Script forms typically denote sets

[‡]Primed script forms typically denote subsets

and the computation performed by a non-input node at time $t = 0$, $A_0 \in \mathcal{V}_0 \backslash \mathcal{V}_{ip}$, as a function f_{A_0} that satisfies

$$f_{A_0}(W(A_0)) = X(\mathcal{Q}(A_0)). \tag{2.5}$$

As before, $W(A_0) \perp\!\!\!\perp W(\mathcal{V}_0 \backslash \{A_0\})$ for all $A_0 \in \mathcal{V}_0$ and $W(\mathcal{V}_0) \perp\!\!\!\perp M$.

Remarks

1. Informally speaking, Definition 2.3 is designed to allow each node to generate a randomized function of its incoming transmissions for each of its outgoing transmissions.

2. The randomization at each node is explicitly captured by its intrinsic random variable $W(\cdot)$, and is assumed to be independent across all nodes of the system.

3. Furthermore, each node is allowed to send a different transmission on each of its outgoing edges.

4. Note that the condition imposed by Equation (2.2) introduces dependence between the random variables in the set $X(\mathscr{E})$.

5. For the most part, we will not be concerned with the precise form of the computation being performed by every node. We will only make use of information-theoretic measures applied to the message and to the random variables in the computational system.

Throughout the paper, we use the variables U, V, A, B, C and D to refer to nodes and E, P, Q, R and S to refer to edges. We use their script forms, e.g. $\mathscr{R}$, when referring to sets of nodes and edges, and primed script forms, e.g. $\mathscr{R}'$, when referring to subsets thereof. Once again, the notation we use is summarized in Table 2.1, and depicted in Figure 2.1 for convenience.

Having defined what we mean by the terms "computational system" and "message", in the following sections we proceed to find a definition for "information flow" and identify properties that this definition satisfies in any computational system.

2.3 Defining Information Flow

Before one can speak of *detecting* information flow in a network, it is first important to *define* what it is that we seek to detect.[13] In this section, we focus on arriving at a definition for information flow.

Our goal is to formalize how information about a message flows in a computational system. Ultimately, we expect to find the *path* that the message takes while being processed by the system. Towards this, we start by trying to formally define what it means for information about the message to flow on a given *edge*. This section concludes with a proposal for such a definition: one based on strict positivity of a conditional mutual information. But to provide the intuition behind this choice of definition, we start with several simpler candidate definitions, and show how they fail to satisfy an intuitive property using counterexamples.

After proposing a definition for information flow, in Section 2.4, we discuss the *properties* satisfied by our definition. Then, in Section 2.5, we specify how the transmissions of the computational system are observed, and describe how information flow might be *inferred* in a real computational system.

2.3.1 An intuitive property

To concretely define what it means for information about a message to flow on an edge, we need some way to assess competing candidate definitions and choose one among them.

[13] In essence, "causal influence" measures such as Granger Causality and Directed Information, while intuitively quantifying transferred information, fail to lay down what *aspect of computation* they actually capture. This is, in part, a result of conflating the stages of defining a quantity we want to understand, and prescribing an estimator for it.

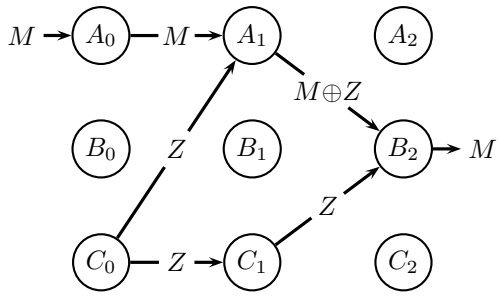


Figure 2.2: The computational system for Counterexample 2.1. We only depict edges relevant to the counterexample here. All other edges in the underlying complete directed graph are still present, but are not shown; their transmissions are assumed to be zero. Observe that no edge at time $t = 1$ has information flow as per Candidate Definition 2.1, yet the message reappears at time $t = 2$.

Towards this goal, we state a straightforward and intuitive property, which we would want any definition of information flow to satisfy.

Suppose that, at a given point in time, there is *no* flow of information about the message across *any* edge of a computational system. Note that this includes self-edges, so no node "carries" information about the message within its memory either. Then, we expect that information about the message has ceased to persist in the system, so the information flow about the message *must* be zero on all edges of the computational system, at all future points in time.

Property 2.1 (The Broken Telephone[14])**.** *Let $\mathscr{C}$ be a computational system, and let $\mathscr{F}_M : \mathscr{C} \to \{0, 1\}$ be an indicator of the presence of information flow about M on an edge. That is, $\mathscr{F}_M(E) = 1$, if information about M flows on the edge $E \in \mathscr{C}$ and $\mathscr{F}_M(E) = 0$, otherwise. The Broken Telephone Property states that if, at some time $t \in \mathscr{T}$, we have*

$$\mathscr{F}_M(E_t) = 0 \quad \forall\, E_t \in \mathscr{C}_t, \tag{2.6}$$

then

$$\mathscr{F}_M(E_{t'}) = 0 \quad \forall\, E_{t'} \in \mathscr{C}_{t'} \;\; \forall\, t' \in \mathscr{T}, t' > t. \tag{2.7}$$

2.3.2 Intuiting Information Flow through Counterexamples

We now propose four candidate definitions, beginning with the simplest. We then construct counterexamples to show how the first three candidate definitions do not satisfy Property 2.1.

Candidate Definition 2.1. *A simplistic and intuitive definition for information flow might simply stem from dependence. We say that information about the message M flows on an edge E_t if*

$$I(M; X(E_t)) > 0.$$

[14] https://en.wikipedia.org/wiki/Telephone_game

Counterexample 2.1. Consider the computational system depicted in Figure 2.2 (note that, in order to avoid unnecessary clutter, only edges with non-zero transmissions are shown in the figure). A_0 is the input node, which has the message $M \sim \text{Ber}(1/2)$ at time $t = 0$. The system is designed to communicate[15] M to the node B using the following strategy: at $t = 0$, A_0 "transmits" M to A_1 (i.e., node A stores M in its memory). C_0 independently generates a different random number, $W(C_0) = Z \sim \text{Ber}(1/2)$, $Z \perp\!\!\!\perp M$, and sends this message to A_1, while also storing it in memory it until $t = 1$. A_1 then computes $M \oplus Z$ and passes the result to B_2, while C_1 sends Z to B_2. Here, the symbol "$\oplus$" stands for XOR, the exclusive-OR operator on two bits. B_2 is thus able to recover M by once again XOR-ing its inputs, $(M \oplus Z)$ and Z.

Note that the output of B_2 depends on M, even though none of its inputs individually depends on M. That is, $I(M; X((A_1, B_2))) = I(M; M \oplus Z) = 0$, and $I(M; X((C_1, B_2))) = I(M; Z) = 0$, so by Candidate Definition 2.1, information about the message flows on *no* edge at time $t = 1$. However, information about the message *does* flow out of node B_2 at time $t = 2$. This violates Property 2.1. Thus, mere *dependence* on the message cannot be a valid definition for flow of information on a single edge. □

Communication strategies such as the one in Counterexample 2.1 frequently arise in cryptography [88], to prevent an eavesdropper from reading confidential information, and in network coding [50], for achieving the communication capacity of a network. Furthermore, a complex computational network may have smaller sub-networks with such topologies. For instance, we observe such a sub-network in the canonical example for network coding: the butterfly network [50, Fig. 7b] (this particular example is discussed in detail in Section 2.6.1). Optimal communication in such a network *requires* the use of such topologies, so Counterexample 2.1 is far from obscure. In fact, central to the idea of Counterexample 2.1 is a concept known as "synergy", which is well-studied in the literature on Partial Information Decomposition [89–91] (see [92] for a recent review). This is discussed at length in Section 2.3.5. Even in neuroscience, the concept of synergy is recognized and well-understood [93–95], and some experimental evidence has appeared in the literature [96].

Counterexample 2.1 demonstrates that the information necessary to recover the message (or a function of it) is not necessarily transmitted through individual edges, but jointly across edges. So, we might instead seek to define the "smallest set of edges" along which information about the message flows, for every point in time. But if we ultimately wish to isolate *paths* along which information about the message flows, we require an understanding of which edges *specifically* the information flows upon. We therefore continue to think of information as flowing on individual edges.[16]

We can now update our naïve definition to counter the previous counterexample. We start by noting that in Counterexample 2.1, although the transmission on edge (A_1, B_2) is

[15] This communication can be thought of as computing the identity function, and making the output available at the node B.

[16] It should be noted that the two views—information flowing on individual edges, versus sets of edges—are compatible with each other if we use Definition 2.5 (which will appear shortly) to describe information flow on a set of edges. This equivalence is elaborated upon in Section 2.3.4. Later, in Section 2.4.4, we attempt to refine our understanding of the aforementioned "smallest set of edges" along which information about the message flows.

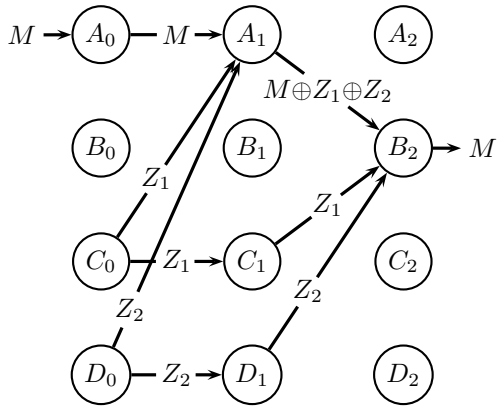


Figure 2.3: The computational system for Counterexample 2.2. Once again, observe that no edge at time $t = 1$ has information flow as per Candidate Definition 2.2, yet the message reappears at time $t = 2$. Note that only edges relevant to the counterexample are depicted in the figure. All other edges of the underlying complete directed graph are still present, and their transmissions are assumed to be zero.

independent of M, it is not *conditionally* independent of M when given the transmission on (C_1, B_2).

Candidate Definition 2.2. *We say that information about the message M flows on an edge $E_t \in \mathscr{E}_t$ if at least one of the following holds:*

1. $I\big(M; X(E_t)\big) > 0$, *or*
2. $\exists\, E'_t \in \mathscr{E}_t$ *s.t.* $I\big(M; X(E_t) \,\big|\, X(E'_t)\big) > 0$.

Counterexample 2.2. Consider a modified version of Counterexample 2.1 in which we XOR M with *two* random variables, Z_1 and Z_2, where $M, Z_1, Z_2 \sim$ i.i.d. Ber$(1/2)$ (as shown in Figure 2.3). Now, since there are two noise terms, no single one of them may be conditioned upon to have non-zero information flow at time $t = 1$. That is, $I(M; M \oplus Z_1 \oplus Z_2 \,|\, Z_1) = 0$ and $I(M; M \oplus Z_1 \oplus Z_2 \,|\, Z_2) = 0$. The same holds true of Z_1, conditioned on either $M \oplus Z_1 \oplus Z_2$ or Z_2, and for Z_2, conditioned on either $M \oplus Z_1 \oplus Z_2$ or Z_1. So, Candidate Definition 2.2 also fails to satisfy Property 2.1. □

It might seem that a possible rectification is to condition on *all* other edges at time t, but we can show that this also fails the test.

Candidate Definition 2.3. *We say that information about the message M flows on an edge $E_t \in \mathscr{E}_t$ if at least one of the following holds:*

1. $I\big(M; X(E_t)\big) > 0$, *or*
2. $I\big(M; X(E_t) \,\big|\, X(\mathscr{E}_t \backslash \{E_t\})\big) > 0$.

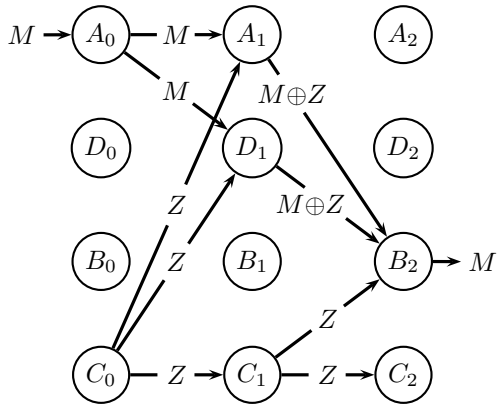


Figure 2.4: The computational system for Counterexample 2.3. Just as in the previous counterexamples, no edge at time $t = 1$ has information flow as per Candidate Definition 2.3, yet the message is reconstructed at time $t = 2$. Note that only edges relevant to the counterexample are depicted in the figure. All other edges of the underlying complete directed graph are still present, and their transmissions are assumed to be zero.

Counterexample 2.3. Consider the computational system shown in Figure 2.4. Once again, we have an input node A_0 which possesses the message at time $t = 0$, and wishes to send this message to node B. It does so by mixing M with an independent random variable Z generated at C_0, so that the scenario described in Counterexample 2.1 still holds. But additionally, A communicates to B along a redundant path, through D_1. Now, if E is any incoming edge of B_2, it is still true that $I(M; X(E)) = 0$. So none of the inputs of B_2 individually depends on M, thus eliminating the first condition in Candidate Definition 2.3. Furthermore, checking each incoming edge of B_2 reveals that the second condition also fails to hold. If we take $E_1 = (A_1, B_2)$, we get

$$I(M; X(E_1) \,|\, X(\mathscr{E}_1 \setminus \{E_1\})) = I(M; M \oplus Z \,|\, M \oplus Z, Z) = 0. \tag{2.8}$$

The same holds true when $E_1 = (D_1, B_2)$ since the transmissions on both edges are identical by construction. Likewise, if we take $E_1 = (C_1, B_2)$, we have

$$I(M; X(E_1) \,|\, X(\mathscr{E}_1 \setminus \{E_1\})) = I(M; Z \,|\, M \oplus Z, Z) = 0, \tag{2.9}$$

with the same holding true when $E_1 = (C_1, C_2)$. Therefore, no edge at time $t = 1$ has any information flow about the message M, as per Candidate Definition 2.3. Nevertheless, B_2 is able to recover the message at time $t = 2$, proving that Property 2.1 fails to hold for Candidate Definition 2.3. □

2.3.3 Information Flow on a Single Edge

The counterexamples presented in the previous section motivate a new definition for when information about the message can be said to flow on a given edge. Neither *dependence* of M on the transmission of an edge, nor conditional dependence given *one* or *all* other edges, satisfy Property 2.1.

However, in all these counterexamples, given an edge E_t upon which we expect to have non-zero information flow, we observe: there is at least one *subset* of edges $\mathscr{E}'_t \subseteq \mathscr{E}_t \backslash \{E_t\}$, such that when given $X(\mathscr{E}'_t)$, $X(E_t)$ is conditionally dependent[17] on M. In Counterexample 2.1, the edge (A_1, B_2), carrying $M \oplus Z$, is conditionally dependent on M, given $X\big((C_1, B_2)\big) = Z$. In Counterexample 2.2, $X\big((A_1, B_2)\big) = M \oplus Z_1 \oplus Z_2$ is conditionally dependent on M, given $\{X\big((C_1, B_2)\big), X\big((D_1, B_2)\big)\} = \{Z_1, Z_2\}$. And finally, in Counterexample 2.3, $X\big((A_1, B_2)\big) = M \oplus Z$ is conditionally dependent on M, given $X\big((C_1, B_2)\big) = Z$; note that we do *not* condition on $X\big((D_1, B_2)\big) = M \oplus Z$. Thus, conditioning on a subset of the other edges' transmissions creates dependence between M and the transmission on an edge of interest.

We will shortly prove that Property 2.1 holds when information flow is defined as below, so we directly state it as a definition, skipping its candidacy status.

Definition 2.4 (M-information Flow on a Single Edge)**.** *We say that information about the message M flows on an edge $E_t \in \mathscr{E}_t$ if*

$$\exists\, \mathscr{E}'_t \subseteq \mathscr{E}_t \setminus \{E_t\} \quad s.t. \quad I\big(M; X(E_t) \,\big|\, X(\mathscr{E}'_t)\big) > 0. \tag{2.10}$$

Henceforth, we refer to "information flow about the message M" as M-information flow*, and use the phrase "the edge E_t* has *M-information flow" or "the edge E_t* carries *M-information flow" to mean that information about M flows on E_t per this definition.*

Note that if $I\big(M; X(E_t) \,\big|\, X(\mathscr{E}'_t)\big) > 0$, then we must have $I\big(M; X(\{E_t\} \cup \mathscr{E}'_t)\big) > 0$. In other words, *there exists* a set of edges that includes E_t, whose transmissions depend on M. This is why it is important to condition on all possible subsets of $\mathscr{E}_t$. It is not immediately clear, however, whether *every* edge in $\{E_t\} \cup \mathscr{E}'_t$ has M-information flow. We return to this point in Section 2.4.4.

Also, this definition implies that certain edges, such as (C_1, B_2) in Counterexample 2.1, may have M-information flow, which may seem counter-intuitive. This is discussed further and justified in Section 2.4.2.

2.3.4 Information Flow on a Set of Edges

The definition of M-information flow for a single edge naturally generalizes to one for a set of edges, at a given time.

Definition 2.5 (M-information Flow on a Set of Edges)**.** *We say that information about the message M flows on a set of edges $\mathscr{E}'_t \subseteq \mathscr{E}_t$ if*

$$\exists\, \mathscr{R}'_t \subseteq \mathscr{E}_t \quad s.t. \quad I\big(M; X(\mathscr{E}'_t) \,\big|\, X(\mathscr{R}'_t)\big) > 0. \tag{2.11}$$

The definition of M-information flow on a set of edges is nearly identical to its single-edge counterpart. Indeed, they are closely related, as the following proposition shows.

Proposition 2.1. *A set $\mathscr{E}'_t \subseteq \mathscr{E}_t$ has M-information flow (per Definition 2.5) if and only if there exists an edge $E'_t \in \mathscr{E}'_t$ that has M-information flow (per Definition 2.4).*

[17] Equivalently, we could say that there exists at least one subset of edges $\mathscr{E}'_t \subseteq \mathscr{E}_t$, without explicitly excluding E_t, since $I\big(M; X(E_t) \,\big|\, X(E_t), X(\mathscr{E}'_t)\big) = 0$.

A proof of this proposition can be found in Appendix 2.A.

It should be noted that although the counterexamples in this section all employed computational systems which recovered the message M at a new node at a later time, a computational system will in general compute some function of the message. For instance, see the example in Section 2.6.2.

2.3.5 The Connection with Synergistic Information

This section connects our definition of M-information flow with recent developments on a subject known as "Partial Information Decomposition" (PID). Our definition is closely related to the concept of "Synergistic Information" that appears in this field. This section exists only for the purpose of providing a deeper intuition for our definition of M-information flow, and does not affect the rest of the paper in any significant way. We have attempted to explain this intuition in a way that is accessible to readers unfamiliar with the PID literature. However, readers may feel free to skip this section, if desired.

At its core, Counterexample 2.1 relies on a concept known as "synergy", which is described explicitly in the literature on Partial Information Decomposition (PID) [89–91] (see [92] for a recent review, and Appendix 2.C for a brief introduction). Essentially, this body of literature seeks to decompose the mutual information that two or more variables share about a message, $I(M;(Y_1,Y_2,\ldots))$, into several individually meaningful, non-negative components. In particular, when discussing the bivariate case—i.e., the case of two variables, $I(M;(Y_1,Y_2))$—it is understood what the terms in this decomposition should be: (i) information about the message that each variable carries *uniquely*, and which cannot be inferred from the other; (ii) information about the message that the variables share *redundantly*, and which can be extracted from either; (iii) and information about the message that the variables convey *synergistically*, which is revealed only when *both* variables are taken together, and cannot be inferred from either variable *individually*. Counterexample 2.1 is the canonical example for synergy, and is known simply as the "XOR" example in the PID literature. While $M \oplus Z$ and Z are *individually* independent of M, when taken *together*, $I(M;(M \oplus Z, Z)) = H(M)$. This suggests that $M \oplus Z$ and Z have no unique or shared information about M, but convey information synergistically.

While the field has not yet arrived at a consensus on the most appropriate definitions for unique, redundant and synergistic information [92], it is well-understood what properties these quantities must satisfy, at least in the bivariate case (see Appendix 2.C, specifically, Equations (2.85), (2.86) and (2.88)). Therefore, even without formal definitions, we can rely on the intuition provided by these properties to understand the implications of PID for M-information flow. If a particular edge's transmission contains unique or redundant information about the message (with respect to some other subset of edges at that point in time), then that information will manifest itself in the form of strictly positive mutual information. However, in the absence of positive mutual information between the message and the transmission on a given edge, we need to consider whether said transmission synergistically interacts with another subset of transmissions at that point in time, as this could potentially create dependence with the message through the kind of "recombination" described in Counterexample 2.1. We then need to decide whether such synergistic interactions ought to be considered to constitute information flow. As we show below, our definition of M-

information flow *does* consider instances of purely synergistic information to constitute information flow.

Indeed, it is possible to formulate a definition for information flow based on synergy, which is completely equivalent to Definition 2.4. The definition below makes use of the PID preliminaries given in Appendix 2.C.

Definition 2.6 (M-synergistic information flow)**.** *We say that an edge E_t has* M-synergistic information flow *if at least one of the following holds:*

1. $I\big(M; X(E_t)\big) > 0$, *or*
2. $\exists\, \mathscr{C}_t' \subseteq \mathscr{C}_t \setminus \{E_t\} \quad s.t. \quad SI\big(M : X(E_t); X(\mathscr{C}_t')\big) > 0,$

where $SI(M : X; Y)$ represents the synergistic information between X and Y about M.

Proposition 2.2 (Equivalence of Information Flow Definitions)**.** *An edge E_t has M-information flow if and only if it has M-synergistic information flow. Furthermore, suppose E_t is an edge which satisfies $I\big(M; X(E_t)\big) = 0$. Then,*

$$I\big(M; X(E_t) \,\big|\, X(\mathscr{C}_t')\big) > 0 \tag{2.12}$$

for some set $\mathscr{C}_t' \subseteq \mathscr{C}_t \setminus \{E_t\}$, if and only if

$$SI\big(M : X(E_t); X(\mathscr{C}_t')\big) > 0. \tag{2.13}$$

That is, the set $\mathscr{C}_t'$ upon whose transmissions we need to condition is the same *as the one responsible for providing synergy in the alternate definition.*

A proof of this proposition is given in Appendix 2.C.

We should also mention here that it may be possible to leverage specific definitions of synergistic information to supply an intuitive measure of the *volume* of information flow; we discuss this in Section 2.7.4.

2.4 Properties of Information Flow

Having defined what it means for information about a message to flow on an edge, we demonstrate that Definition 2.4 satisfies several intuitively desirable properties, including Property 2.1.

2.4.1 The Broken Telephone Property

Theorem 2.3. *M-information flow satisfies the Broken Telephone Property, i.e., Definition 2.4 satisfies Property 2.1.*

Before we prove this theorem, we prove a simpler lemma which directly falls out of Definition 2.4 and the properties of mutual information.

Lemma 2.4. *There is* no *edge in $\mathscr{E}_t$ that carries M-information flow if, and only if, $X(\mathscr{E}_t)$ is independent of M. In other words,*

$$I(M; X(E_t) \,|\, X(\mathscr{E}'_t)) = 0 \quad \forall\, E_t \in \mathscr{E}_t,\ \mathscr{E}'_t \subseteq \mathscr{E}_t \backslash \{E_t\} \tag{2.14}$$

if and only if

$$I(M; X(\mathscr{E}_t)) = 0. \tag{2.15}$$

Equivalently, we can state the opposite: $X(\mathscr{E}_t)$ depends *on M if and only if* at least *one edge in $\mathscr{E}_t$ carries M-information flow.*

Proof. ($\Rightarrow$) Suppose that the condition in (2.14) holds. Let $\mathscr{E}_t = \{E_t^{(1)}, E_t^{(2)}, \dots, E_t^{(N^2)}\}$ be any ordering of the edges in $\mathscr{E}_t$. Then,

$$\begin{aligned} I(M; X(\mathscr{E}_t)) &\overset{(a)}{=} I(M; X(E_t^{(1)})) + I(M; X(E_t^{(2)}) \,|\, X(E_t^{(1)})) && (2.16)\\ &\quad + I(M; X(E_t^{(3)}) \,|\, X(E_t^{(1)}), X(E_t^{(2)})) + \cdots \\ &= \sum_{i=1}^{N^2} I\Big(M; X(E_t^{(i)}) \,\Big|\, \bigcup_{j=1}^{i-1} \{X(E_t^{(j)})\}\Big) && (2.17)\\ &\overset{(b)}{=} \sum_{i=1}^{N^2} I\Big(M; X(E_t^{(i)}) \,\Big|\, X\Big(\bigcup_{j=1}^{i-1} \{E_t^{(j)}\}\Big)\Big) \overset{(c)}{=} 0, && (2.18) \end{aligned}$$

where (a) follows from the chain-rule of mutual information [97, Ch. 2], (b) is simply the application of Equation (2.1), and (c) follows from the fact that each term in the summation is zero, by (2.14). This proves the forward implication.

($\Leftarrow$) Next, suppose $I(M; X(\mathscr{E}_t)) = 0$. Let E_t be any edge in $\mathscr{E}_t$ and let $\mathscr{E}'_t$ be any subset of $\mathscr{E}_t \setminus \{E_t\}$. Also, let $\mathscr{E}''_t = \mathscr{E}_t \setminus (\mathscr{E}'_t \cup \{E_t\})$. Then,

$$\begin{aligned} 0 &= I(M; X(\mathscr{E}_t)) && (2.19)\\ &= I(M; X(\mathscr{E}'_t)) + I(M; X(E_t) \,|\, X(\mathscr{E}'_t)) + I(M; X(\mathscr{E}''_t) \,|\, X(\mathscr{E}'_t), X(E_t)) && (2.20) \end{aligned}$$

by the chain rule. Since (conditional) mutual information is always non-negative [97, Ch. 2], all three terms on the right hand side must be zero. So in particular,

$$I(M; X(E_t) \,|\, X(\mathscr{E}'_t)) = 0. \tag{2.21}$$

Since E_t and $\mathscr{E}'_t$ are arbitrary, this proves the converse. □

Proof of Theorem 2.3. We need to prove that M-information flow, as given by Definition 2.4, satisfies Property 2.1. Explicitly stated, we need to show that if every edge at some time t has zero M-information flow, then every edge at all future times $t' > t$ must also have zero M-information flow. So suppose that, at time t, for every $E_t \in \mathscr{E}_t$ we have

$$I(M; X(E_t) \,|\, X(\mathscr{E}'_t)) = 0 \quad \forall\, \mathscr{E}'_t \subseteq \mathscr{E}_t \setminus \{E_t\}. \tag{2.22}$$

By Lemma 2.4, this implies that

$$I(M; X(\mathscr{C}_t)) = 0. \tag{2.23}$$

Now, consider the first future time instant, $t' = t + 1$. For every node $A_{t+1} \in \mathscr{V}_{t+1}$, the definition of computation at a node (Definition 2.3b) states that

$$X(\mathbb{Q}(A_{t+1})) = f_{A_{t+1}}(X(\mathscr{P}(A_{t+1})), W(A_{t+1})), \tag{2.24}$$

where the reader may recall, $\mathscr{P}(A_{t+1})$ and $\mathbb{Q}(A_{t+1})$ are the edges entering and leaving A_{t+1} respectively. We can collect the individual functions $f_{A_{t+1}}$ across all nodes in $\mathscr{V}_{t+1}$ into a single joint function $f_{\mathscr{V}_{t+1}}$, as described in Definition 2.3b, to obtain

$$X(\mathscr{C}_{t+1}) = f_{\mathscr{V}_{t+1}}(X(\mathscr{C}_t), W(\mathscr{V}_{t+1})). \tag{2.25}$$

Therefore,

$$\begin{aligned}
0 \overset{(a)}{\leq} I(M; X(\mathscr{C}_{t+1})) &= I(M; f_{\mathscr{V}_{t+1}}(X(\mathscr{C}_t), W(\mathscr{V}_{t+1}))) && (2.26)\\
&\overset{(b)}{\leq} I(M; X(\mathscr{C}_t), W(\mathscr{V}_{t+1})) && (2.27)\\
&= I(M; X(\mathscr{C}_t)) + I(M; W(\mathscr{V}_{t+1}) \,|\, X(\mathscr{C}_t)) && (2.28)\\
&\overset{(c)}{=} I(M; X(\mathscr{C}_t)) \overset{(d)}{=} 0, && (2.29)
\end{aligned}$$

where (a) follows from the non-negativity of mutual information, (b) is an application of the Data Processing Inequality [97, Ch. 2], (c) follows from the fact that $W(\mathscr{V}_{t+1}) \perp\!\!\!\perp \{M, X(\mathscr{C}_t)\}$, as stated in Definition 2.3b, and (d) follows from (2.23). So, we must have that $I(M; X(\mathscr{C}_{t+1})) = 0$. Applying Lemma 2.4 once again, we find that for $t' = t + 1$,

$$I(M; X(E_{t'}) \,|\, X(\mathscr{C}'_{t'})) = 0 \quad \forall\, E_{t'} \in \mathscr{C}_{t'}, \mathscr{C}'_{t'} \subseteq \mathscr{C}_{t'} \backslash \{E_{t'}\} \tag{2.30}$$

We have shown that (2.22) implies (2.30), so induction on t' yields that (2.30) holds for all future times $t' > t$, completing the proof. □

2.4.2 The Existence of Orphans

M-information flow (Definition 2.4) also has a very non-intuitive property: an edge leading out of a node may have M-information flow, even though *no* edge leading *into* that node has M-information flow.

Definition 2.7 (M-information Orphan)**.** *In a computational system $\mathscr{C}$, a node V_t is said to be an* M-information orphan *if $\mathbb{Q}(V_t)$ has M-information flow (as per Definition 2.5), but $\mathscr{P}(V_t)$ has no M-information flow.*

Property 2.2. *M-information orphans may exist in a computational system.*

Proof. Consider the computational system in Figure 2.2 from Counterexample 2.1. The node C_1 is an M-information orphan, since its outgoing edge (C_1, B_2) carries M-information flow, whereas none of its incoming edges carries M-information flow. □

The existence of M-information orphans, along with the presence of M-information flow on (C_1, B_2) in Counterexample 2.1, may not be expected, since Z was never computed from M. Indeed, M-information flow appears to emerge from "nowhere" at the node C_1, leaving it *orphaned* in a view of the graph that contains only edges having M-information flow (hence the name). But closer inspection reveals that in this example, the transmissions arriving at B_2 from A_1 and C_1, i.e. $M \oplus Z$ and Z, are *statistically identical*: they are both individually independent of M, but when XOR'ed, are fully dependent on M. In other words, any *purely observational* measure[18] defined on the transmissions at time t that assigns M-information flow to $M \oplus Z$, must also assign M-information flow to Z.

Note that, just as M-information flow can originate at an M-information orphan, M-information flow may also terminate at a node—either by simple omission, or as a result of some computation (see Section 2.6 for such instances). Likewise, multiple outgoing edges of a given node may transmit redundant copies of the same information. Ultimately, we see that there is no "law of conservation" for M-information flow. In this sense, "information flow" is not a typical kind of "flow" that is defined on graphs (see, for example, [98, Sec. 26.1]), and well-known results such as the Max-flow Min-cut Theorem [98, Thm. 26.6] do not apply as-is to M-information flow.

It is worthwhile to note at this point that the existence of M-information orphans such as C_1 in Counterexample 2.1 is not inconsistent with the Data Processing Inequality [97, Ch. 2]. In fact, a clear example of the Data Processing Inequality is seen at the network-level, wherein M—$X(\mathscr{E}_t)$—$X(\mathscr{E}_{t+1})$ form a Markov Chain for any time $0 \leq t < T$, and so the information content about M present collectively in all transmissions at time $t+1$ *must* be no more than that present at time t. We call this Global Markovity, and state it formally for completeness.

Corollary 2.5 (Global Markovity)**.** *At any given time t, the following Markov Chain holds:* M—$X(\mathscr{E}_t)$—$X(\mathscr{E}_{t+1})$.

In fact, this Markov condition must hold for every *subset* of nodes, not just for the entire set of nodes, so it is subsumed by the following proposition.

Proposition 2.6 (Local Markovity)**.** *At any time t, for any given subset of nodes $\mathscr{V}_t' \subseteq \mathscr{V}_t$, the following Markov Chain holds:* M—$X(\mathscr{P}(\mathscr{V}_t'))$—$X(\mathbb{Q}(\mathscr{V}_t'))$.

Proof. Since $X(\mathbb{Q}(\mathscr{V}_t')) = f_{\mathscr{V}_t'}(X(\mathscr{P}(\mathscr{V}_t')), W(\mathscr{V}_t'))$ by Definition 2.3b, the tuple $(X(\mathscr{P}(\mathscr{V}_t')), X(\mathbb{Q}(\mathscr{V}_t')))$ is also a function of $X(\mathscr{P}(\mathscr{V}_t'))$ and $X(W(\mathscr{V}_t'))$. Hence, the following Markov chain holds:

$$M\text{—}(X(\mathscr{P}(\mathscr{V}_t')), W(\mathscr{V}_t'))\text{—}(X(\mathscr{P}(\mathscr{V}_t')), X(\mathbb{Q}(\mathscr{V}_t'))).$$

[18] i.e., a functional of the joint distribution of $X(\mathscr{E}_t)$

By the Data Processing Inequality, this implies that

$$I(M; X(\mathbb{Q}(\mathscr{V}_t')), X(\mathscr{P}(\mathscr{V}_t'))) \leq I(M; X(\mathscr{P}(\mathscr{V}_t')), W(\mathscr{V}_t')) \tag{2.31}$$

$$\overset{(a)}{=} I(M; X(\mathscr{P}(\mathscr{V}_t'))) + I(M; W(\mathscr{V}_t') \,|\, X(\mathscr{P}(\mathscr{V}_t'))) \tag{2.32}$$

$$\overset{(b)}{=} I(M; X(\mathscr{P}(\mathscr{V}_t'))) + I(W(\mathscr{V}_t'); M, X(\mathscr{P}(\mathscr{V}_t'))) \tag{2.33}$$

$$- I(W(\mathscr{V}_t'); X(\mathscr{P}(\mathscr{V}_t')))$$

$$\overset{(c)}{=} I(M; X(\mathscr{P}(\mathscr{V}_t'))) + 0 - 0, \tag{2.34}$$

where in (a) and (b), we have used the chain rule of mutual information in two different ways, and in (c) we have used the fact that $W(\mathscr{V}_t') \perp\!\!\!\perp \{M, X(\mathscr{P}(\mathscr{V}_t'))\}$. Therefore,

$$I(M; X(\mathbb{Q}(\mathscr{V}_t')) \,|\, X(\mathscr{P}(\mathscr{V}_t'))) = 0, \tag{2.35}$$

which implies the Markov chain in Proposition 2.6. □

Since the above also holds for $\mathscr{V}_t' = \mathscr{V}_t$, wherein $\mathbb{Q}(\mathscr{V}_t) = \mathscr{E}_t$, Proposition 2.6 implies Corollary 2.5.

Given that these Markov conditions arise directly from the way we have defined the computational system, specifically Definition 2.3b, they may not be very surprising (indeed, they may be considered *properties* of the computational system model itself). However, it is worth noting that Proposition 2.6 holds *even at an M-information orphan*. Thus, M-information orphans do not "create" information about M, as we would rightly expect, given the Data Processing Inequality.

2.4.3 The Existence of Information Paths

We now show that if the outgoing transmissions of any given node depend on the message, then we can find a path leading to that node from one or more input nodes, along which M-information flows. Before we demonstrate this property, we formally define what we mean by the terms "path" and "cut".

Definition 2.8 (Path)**.** *In any computational system $\mathscr{C}$, suppose $\mathscr{A}$ and $\mathscr{B}$ are two disjoint sets of nodes in $\mathscr{V}$. Then, a* path *from $\mathscr{A}$ to $\mathscr{B}$ is any ordered set of nodes $\{V^{(0)}, V^{(1)}, \ldots, V^{(L)}\}$ that satisfies (i) $V^{(0)} \in \mathscr{A}$; (ii) $V^{(L)} \in \mathscr{B}$; and (iii) $(V^{(i-1)}, V^{(i)}) \in \mathscr{E}$ for every $1 \leq i \leq L$, where L is a positive integer indicating the path's length. We refer to the set $\{(V^{(i-1)}, V^{(i)})\}_{i=1}^{L}$ as the* edges of the path.

Definition 2.9 (M-Information Path)**.** *Continuing from Definition 2.8, we define an M-*information path *from $\mathscr{A}$ to $\mathscr{B}$ as any path from $\mathscr{A}$ to $\mathscr{B}$, each of whose edges carries M-information flow. That is, if $(V^{(i-1)}, V^{(i)}) = E_{t_i} \in \mathscr{E}_{t_i}$ for some $t_i \in \mathscr{T}$, then for every $1 \leq i \leq L$,*

$$\exists\, \mathscr{E}_{t_i}' \subseteq \mathscr{E}_{t_i} \quad s.t. \quad I(M; X(E_{t_i}) \,|\, X(\mathscr{E}_{t_i}')) > 0. \tag{2.36}$$

Definition 2.10 (Cut)**.** *In any computational system $\mathscr{C}$, suppose $\mathscr{A}$ and $\mathscr{B}$ are two disjoint sets of nodes in $\mathscr{V}$. Then, a* cut *separating $\mathscr{A}$ and $\mathscr{B}$ is any pair of sets $(\mathscr{V}^{src}, \mathscr{V}^{sink})$, such that (i) $\mathscr{V}^{src} \cup \mathscr{V}^{sink} = \mathscr{V}$; (ii) $\mathscr{V}^{src} \cap \mathscr{V}^{sink} = \emptyset$; (iii) $\mathscr{A} \subseteq \mathscr{V}^{src}$; and (iv) $\mathscr{B} \subseteq \mathscr{V}^{sink}$. We refer to the set of edges* going from $\mathscr{V}^{\text{src}}$ to $\mathscr{V}^{\text{sink}}$, *i.e. $\mathscr{E} \cap (\mathscr{V}^{src} \times \mathscr{V}^{sink})$, as the* edges in the cut set.[19]

Definition 2.11 (Zero–M-information Cut)**.** *Continuing from Definition 2.10, we say that a cut $(\mathscr{V}^{src}, \mathscr{V}^{sink})$ is a* zero–M-information cut *if every edge in its cut set has zero M-information flow. That is, for every $E_t \in \mathscr{E} \cap (\mathscr{V}^{src} \times \mathscr{V}^{sink})$,*

$$I(M; X(E_t) \,|\, X(\mathscr{E}'_t)) = 0 \quad \forall\, \mathscr{E}'_t \subseteq \mathscr{E}_t \setminus \{E_t\}. \tag{2.37}$$

Remark In Definition 2.11, we require that Equation (2.37) hold for every edge E_t in $\mathscr{E} \cap (\mathscr{V}^{\text{src}} \times \mathscr{V}^{\text{sink}})$. However, the edges in this set may belong to several different time instants, since the cut is not restricted to any particular time (e.g., see Figure 2.5). The time t used in Equation (2.37), therefore, is determined by the time of the edge E_t, and varies for each E_t that we check in $\mathscr{E} \cap (\mathscr{V}^{\text{src}} \times \mathscr{V}^{\text{sink}})$.

Property 2.3 (Existence of an Information Path)**.** *In any computational system $\mathscr{C}$, suppose that at some time $t_{op} \in \mathscr{T}$, there is an "output node" $V_{op} \in \mathscr{V}$ whose outgoing edges $\mathbb{Q}(V_{op})$ satisfy $I(M; X(\mathbb{Q}(V_{op}))) > 0$. Then, there must exist an M-information path from the input nodes $\mathscr{V}_{ip}$ to V_{op}.*

Theorem 2.7. *Definition 2.4 satisfies Property 2.3.*

Informally put, Theorem 2.7 states that our definition of M-information flow (Definition 2.4) guarantees M-information paths to every output node whose outgoing transmissions depend on the message. While the theorem seems obvious on the surface, the proof is in fact non-trivial because of the nature of our definition of M-information flow. Due to Property 2.2, M-information flowing *out of* a node does *not* imply that M-information must flow *into* that node. Therefore, a straightforward application of the Data Processing Inequality at every node fails to prove the theorem, and we must resort to a more rigorous cut-set-based approach.

Proof outline. We shall prove the contrapositive of the theorem, i.e., we will show that if there exists no M-information path from $\mathscr{V}_{\text{ip}}$ to V_{op}, then the outgoing transmissions of V_{op} are independent of M. We first connect the absence of any M-information path with the presence of a zero–M-information cut. This is achieved in Lemma 2.8, which we present before the proof of Theorem 2.7.

The proof itself proceeds by induction over time. We divide the proof into two steps: initialization and continuation. Starting with the first nodes that come after the cut (temporally) in the initialization step, we systematically show that all nodes to the right

[19] Note that it is not necessary for us to assume that, individually, $\mathscr{V}^{\text{src}}$ and $\mathscr{V}^{\text{sink}}$ are *connected* sets of nodes. For instance, there may be an isolated subset of $\mathscr{V}^{\text{sink}}$, surrounded only by nodes in $\mathscr{V}^{\text{src}}$. Our theorems and proofs remain unaffected, even in such a scenario.

of the cut have outgoing transmissions that are independent of the message M through induction. In this proof outline, we show these steps intuitively using Figure 2.5, where the dashed black line denotes the cut.

Initialization. Here, node C_1 is the first node to the right of the cut, and all of its incoming edges must come from across the cut (depicted by lines in red). Because the cut is a zero–M-information cut, none of its incoming transmissions have M-information flow. Furthermore, the intrinsically generated random variable $W(C_1)$ is independent of M and all past transmissions. Using these two facts along with the Data Processing Inequality, we can show that the transmissions on C_1's outgoing edges, $X(\mathbb{Q}(C_1))$, are also independent of M.

Continuation. At the second time instant to the right of the cut, nodes B_2 and C_2 receive their incoming transmissions from either C_1 (shown in orange) or from across the cut (shown in blue). Once again, the transmissions coming from across the cut can have no information flow, and we have shown that the transmissions coming from C_1 are independent of M. Also, $W(B_2)$ and $W(C_2)$ are independent of M and all incoming transmissions. This suffices to show that the outgoing transmissions of B_2 and C_2, $X(\mathbb{Q}(B_2) \cup \mathbb{Q}(C_2))$, are independent of M. Applying this argument repeatedly over time shows that the transmissions of all nodes to the right of the cut are independent of M.

Therefore, if there is a node V_{op} whose outputs depend on M, we can be assured that there exists no zero–M-information cut separating $\mathscr{V}_{\text{ip}}$ from V_{op}. Therefore, by Lemma 2.8, there exists an M-information path from $\mathscr{V}_{\text{ip}}$ to V_{op}. □

A few nuances are omitted in this outline, such as how the definition of $\mathscr{V}_{\text{ip}}$ plays a role precisely. These subtleties are better elucidated in the full proof.

Before proceeding to the formal proof of Theorem 2.7, we first state and prove the lemma we alluded to earlier, which shows how the absence of an M-information path implies the presence of a zero–M-information cut, and vice versa.

Lemma 2.8. *Let $\mathscr{A}$ and $\mathscr{B}$ be two disjoint sets of nodes in the computational system $\mathscr{C}$. There exists no M-information path from $\mathscr{A}$ to $\mathscr{B}$ if and only if there is a zero–M-information cut separating $\mathscr{A}$ and $\mathscr{B}$.*

Proof. ($\Rightarrow$) Suppose there exists no M-information path from $\mathscr{A}$ to $\mathscr{B}$. Consider the set of all nodes to which there exists at least one M-information path from $\mathscr{A}$. Let $\mathscr{V}^{\text{src}}$ be the collection of all such nodes, along with the nodes in $\mathscr{A}$, i.e.,

$$\mathscr{V}^{\text{src}} \coloneqq \mathscr{A} \cup \{V_t \in \mathscr{V} : \exists \text{ an } M\text{-information path from } \mathscr{A} \text{ to } V_t\}. \tag{2.38}$$

Let $\mathscr{V}^{\text{sink}} = \mathscr{V} \setminus \mathscr{V}^{\text{src}}$, so that $\mathscr{V}^{\text{sink}}$ consists of nodes to which there is no M-information path from $\mathscr{A}$. Then, we must have $\mathscr{B} \subseteq \mathscr{V}^{\text{sink}}$, since it is known that there are no M-information paths from $\mathscr{A}$ to $\mathscr{B}$. Therefore, $(\mathscr{V}^{\text{src}}, \mathscr{V}^{\text{sink}})$ is a cut that separates $\mathscr{A}$ and $\mathscr{B}$, such that no edge in the cut set has M-information flow. In other words, by Definition 2.11, this is a zero–M-information cut separating $\mathscr{A}$ and $\mathscr{B}$.

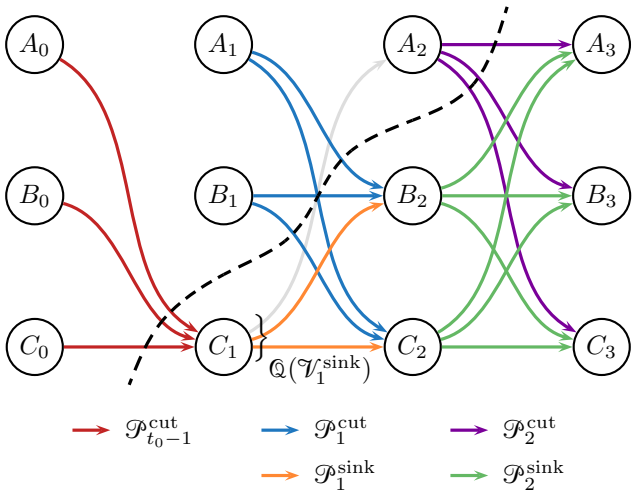


Figure 2.5: A generic computational system used in the proof outline and to explain certain steps in the proof of Theorem 2.7. For the purposes of the proof outline, it suffices to note that the black dashed line denotes the cut. All variable names can be ignored at this point of time.
For the purposes of the formal proof, note that in this figure, $\mathcal{E}^{\text{cut}}$ is essentially the union of the red, blue and purple edges, while $\mathcal{E}^{\text{sink}}$ is the union of the orange and green edges. From this, it is evident that $\mathcal{P}(\mathcal{V}_t^{\text{sink}}) = \mathcal{P}_{t-1}^{\text{cut}} \cup \mathcal{P}_{t-1}^{\text{sink}}$ for any time t, i.e., the incoming edges of $\mathcal{V}^{\text{sink}}$ at time t must either come from nodes in $\mathcal{V}^{\text{sink}}$ or from nodes across the cut. Secondly, it should be clear that $\mathcal{P}_{t-1}^{\text{sink}} = \mathbb{Q}(\mathcal{V}_{t-1}^{\text{sink}}) \cap \mathcal{E}^{\text{sink}}$, i.e., the *incoming* edges of $\mathcal{V}_t^{\text{sink}}$ that originate from nodes in $\mathcal{V}^{\text{sink}}$ are simply the *outgoing* edges of $\mathcal{V}_{t-1}^{\text{sink}}$ which terminate at nodes in $\mathcal{V}^{\text{sink}}$. This is seen best at time $t = 1$ in the graph above, where the orange and grey lines together represent $\mathbb{Q}(\mathcal{V}_1^{\text{sink}})$, the orange and green edges together make up $\mathcal{E}^{\text{sink}}$, and $\mathcal{P}_1^{\text{sink}}$ is given by the orange edges, which is the intersection of the two sets.

($\Leftarrow$) Next, suppose that there *is* an M-information path $\{V^{(i)}\}_{i=0}^{L}$ from $\mathcal{A}$ to $\mathcal{B}$. Then, we claim that there can exist no zero–M-information cut separating $\mathcal{A}$ and $\mathcal{B}$. Let $(\mathcal{V}^{\text{src}}, \mathcal{V}^{\text{sink}})$ be any cut separating $\mathcal{A}$ and $\mathcal{B}$. By Definitions 2.8 and 2.10, we must have $V^{(0)} \in \mathcal{A} \subseteq \mathcal{V}^{\text{src}}$ and $V^{(L)} \in \mathcal{B} \subseteq \mathcal{V}^{\text{sink}}$. So, there must be at least one edge going from $\mathcal{V}^{\text{src}}$ to $\mathcal{V}^{\text{sink}}$ which lies on the path. This implies that at least one edge in the cut set carries M-information flow. Since the conditions of Definition 2.11 are not satisfied, this cut is *not* a zero–M-information cut. Since this is true for every cut separating $\mathcal{A}$ and $\mathcal{B}$, the claim holds. □

Proof of Theorem 2.7. As mentioned in the proof outline, we prove the contrapositive of the theorem. Suppose there exists no M-information path from the input nodes $\mathcal{V}_{\text{ip}}$ to V_{op}. Then, by Lemma 2.8, there exists a zero–M-information cut[20] separating $\mathcal{V}_{\text{ip}}$ and V_{op}. We use this to prove that the transmissions of V_{op} are independent of M.

Setup. Let the zero–M-information cut separating $\mathcal{V}_{\text{ip}}$ and V_{op} be given by $(\mathcal{V}^{\text{src}}, \mathcal{V}^{\text{sink}})$, so that $\mathcal{V}_{\text{ip}} \subseteq \mathcal{V}^{\text{src}}$ and $V_{\text{op}} \in \mathcal{V}^{\text{sink}}$. Then, the cut divides $\mathcal{E}$ into the following sets:

1. $\mathcal{E}^{\text{src}} := \mathcal{E} \cap (\mathcal{V}^{\text{src}} \times \mathcal{V}^{\text{src}})$, the edges between the nodes in $\mathcal{V}^{\text{src}}$;

2. $\mathcal{E}^{\text{sink}} := \mathcal{E} \cap (\mathcal{V}^{\text{sink}} \times \mathcal{V}^{\text{sink}})$, the edges between the nodes in $\mathcal{V}^{\text{sink}}$; and

[20] Note that, in general, this cut may be arbitrarily complex, spanning several nodes and multiple time instants.

3. $\mathscr{E}^{\text{cut}} := \mathscr{E} \cap (\mathscr{V}^{\text{src}} \times \mathscr{V}^{\text{sink}})$, the edges going from $\mathscr{V}^{\text{src}}$ to $\mathscr{V}^{\text{sink}}$.

(Note that the edges going from $\mathscr{V}^{\text{sink}}$ to $\mathscr{V}^{\text{src}}$ will not be relevant to our discussion). As stated before, Lemma 2.8 implies that $(\mathscr{V}^{\text{src}}, \mathscr{V}^{\text{sink}})$ is a zero–M-information cut, so by Definition 2.11, we have that for all $E_t \in \mathscr{E}^{\text{cut}}$,

$$I(M; X(E_t) \,|\, X(\mathscr{E}'_t)) = 0 \quad \forall\, \mathscr{E}'_t \subseteq \mathscr{E}_t \setminus \{E_t\}. \tag{2.39}$$

Note that the edges in $\mathscr{E}^{\text{cut}}$ may belong to different time instants. In particular, the time instant t in the equation above corresponds to the time of the edge E_t, whose flow is in question.[21]

Order the nodes in $\mathscr{V}^{\text{sink}}$ by time, and let $\mathscr{V}_t^{\text{sink}}$ be the subset of nodes in $\mathscr{V}^{\text{sink}}$ at time t. Let $\mathscr{P}(\mathscr{V}_t^{\text{sink}})$ and $\mathbb{Q}(\mathscr{V}_t^{\text{sink}})$ respectively be the sets of edges *collectively* entering and leaving all nodes in $\mathscr{V}_t^{\text{sink}}$. We shall prove that the outgoing transmissions of every node in $\mathscr{V}^{\text{sink}}$, including those of V_{op}, must be independent of the message, i.e.,

$$I(M; X(\mathbb{Q}(V))) = 0 \quad \forall\, V \in \mathscr{V}^{\text{sink}}. \tag{2.40}$$

Initialization. Let t_0 be the first time instant t for which $\mathscr{V}_t^{\text{sink}}$ is non-empty. Then, we encounter two cases: either $t_0 = 0$, in which case the nodes in $\mathscr{V}_{t_0}^{\text{sink}}$ have *no* incoming edges, or $t_0 > 0$, and the nodes in $\mathscr{V}_{t_0}^{\text{sink}}$ *have* incoming edges. We shall first prove that in *both* cases, the outgoing transmissions of $\mathscr{V}_{t_0}^{\text{sink}}$ are independent of the message, i.e. $I(M; X(\mathbb{Q}(\mathscr{V}_{t_0}^{\text{sink}}))) = 0$.

(Case I) When $t_0 = 0$, $\mathscr{V}_0^{\text{sink}} \cap \mathscr{V}_{\text{ip}} = \emptyset$. This is because the cut separates $\mathscr{V}_{\text{ip}}$ from V_{op}, with $\mathscr{V}_{\text{ip}} \subseteq \mathscr{V}^{\text{src}}$, so no nodes in $\mathscr{V}_0^{\text{sink}}$ can be input nodes. So, by the definition of (non-)input nodes (Definition 2.3c), we must have

$$I(M; X(\mathbb{Q}(\mathscr{V}_0^{\text{sink}}))) = I(M; f_{\mathscr{V}_0^{\text{sink}}}(W(\mathscr{V}_0^{\text{sink}}))) \tag{2.41}$$

$$\overset{(a)}{\leq} I(M; W(\mathscr{V}_0^{\text{sink}})) \tag{2.42}$$

$$\overset{(b)}{=} 0, \tag{2.43}$$

where step (a) uses the data processing inequality and step (b) makes use of the fact that $W(\mathscr{V}_0) \perp\!\!\!\perp M$.

(Case II) When $t_0 > 0$, the definition of t_0 implies that all nodes at time $t_0 - 1$ are in $\mathscr{V}^{\text{src}}$, so all incoming edges of $\mathscr{V}_{t_0}^{\text{sink}}$ must lie in the cut set, i.e., $\mathscr{P}(\mathscr{V}_{t_0}^{\text{sink}}) \subseteq \mathscr{E}^{\text{cut}}$. Since the cut is a zero–M-information cut, we have that for all $E_{t_0-1} \in \mathscr{P}(\mathscr{V}_{t_0}^{\text{sink}})$,

$$I(M; X(E_{t_0-1}) \,|\, X(\mathscr{E}'_{t_0-1})) = 0 \quad \forall\, \mathscr{E}'_{t_0-1} \subseteq \mathscr{E}_{t_0-1}. \tag{2.44}$$

By the definition of M-information flow for a set of edges (Definition 2.5) and Proposition 2.1, we have

$$I(M; X(\mathscr{P}(\mathscr{V}_{t_0}^{\text{sink}})) \,|\, X(\mathscr{E}'_{t_0-1})) = 0 \quad \forall\, \mathscr{E}'_{t_0-1} \subseteq \mathscr{E}_{t_0-1}. \tag{2.45}$$

[21] This is one of the complicating factors that prevents us from recursively applying the Data Processing Inequality at every node, to trace a path backwards from V_{op} to $\mathscr{V}_{\text{ip}}$.

Once again, considering $\mathbb{Q}(\mathcal{V}_{t_0}^{\text{sink}})$, we have

$$\begin{aligned} I(M; X(\mathbb{Q}(\mathcal{V}_{t_0}^{\text{sink}}))) &= I(M; f_{\mathcal{V}_{t_0}^{\text{sink}}}(X(\mathcal{P}(\mathcal{V}_{t_0}^{\text{sink}})), W(\mathcal{V}_{t_0}^{\text{sink}}))) && (2.46)\\ &\overset{(a)}{\leq} I(M; X(\mathcal{P}(\mathcal{V}_{t_0}^{\text{sink}})), W(\mathcal{V}_{t_0}^{\text{sink}})) && (2.47)\\ &\overset{(b)}{=} I(M; X(\mathcal{P}(\mathcal{V}_{t_0}^{\text{sink}}))) + I(M; W(\mathcal{V}_{t_0}^{\text{sink}}) \,|\, X(\mathcal{P}(\mathcal{V}_{t_0}^{\text{sink}}))) && (2.48)\\ &\overset{(c)}{=} 0, && (2.49) \end{aligned}$$

where (a) and (b) follow from the Data Processing Inequality and the chain rule of mutual information respectively. In step (c), the first expression in the sum goes to zero by taking $\mathcal{E}_{t_0-1} = \emptyset$ in (2.45) and the second expression is zero since $W(\mathcal{V}_{t_0}^{\text{sink}}) \perp\!\!\!\perp \{M, X(\mathcal{E}_{t_0-1})\}$, and $\mathcal{P}(\mathcal{V}_{t_0}^{\text{sink}}) \subseteq \mathcal{E}_{t_0-1}$ (refer Definition 2.3b). So, from equations (2.43) and (2.49), we have that for all values of t_0,

$$I(M; X(\mathbb{Q}(\mathcal{V}_{t_0}^{\text{sink}}))) = 0. \tag{2.50}$$

Continuation. Now, suppose that for some $t > t_0$, we have $I(M; X(\mathbb{Q}(\mathcal{V}_{t-1}^{\text{sink}}))) = 0$. We shall prove that this implies $I(M; X(\mathbb{Q}(\mathcal{V}_{t}^{\text{sink}}))) = 0$. First, observe that

$$\mathcal{P}(\mathcal{V}_t^{\text{sink}}) = (\mathcal{P}(\mathcal{V}_t^{\text{sink}}) \cap \mathcal{E}^{\text{cut}}) \cup (\mathcal{P}(\mathcal{V}_t^{\text{sink}}) \cap \mathcal{E}^{\text{sink}}) \tag{2.51}$$

For convenience, let $\mathcal{P}_{t-1}^{\text{cut}} \coloneqq \mathcal{P}(\mathcal{V}_t^{\text{sink}}) \cap \mathcal{E}^{\text{cut}}$ and $\mathcal{P}_{t-1}^{\text{sink}} \coloneqq \mathcal{P}(\mathcal{V}_t^{\text{sink}}) \cap \mathcal{E}^{\text{sink}}$. We have used the subscript $t-1$ here to remind the reader that $\mathcal{P}(\mathcal{V}_t^{\text{sink}})$, which are the *incoming* edges of $\mathcal{V}_t^{\text{sink}}$, are a subset of $\mathcal{E}_{t-1}$. Then, we have

$$\mathcal{P}(\mathcal{V}_t^{\text{sink}}) = \mathcal{P}_{t-1}^{\text{cut}} \cup \mathcal{P}_{t-1}^{\text{sink}}. \tag{2.52}$$

Since the cut is a zero–M-information cut, we have that for every $E_{t-1} \in \mathcal{P}_{t-1}^{\text{cut}}$,

$$I(M; X(E_{t-1}) \,|\, X(\mathcal{E}'_{t-1})) = 0 \quad \forall\, \mathcal{E}'_{t-1} \subseteq \mathcal{E}_{t-1}. \tag{2.53}$$

Therefore, by Definition 2.5 and Proposition 2.1,

$$I(M; X(\mathcal{P}_{t-1}^{\text{cut}}) \,|\, X(\mathcal{E}'_{t-1})) = 0 \quad \forall\, \mathcal{E}'_{t-1} \subseteq \mathcal{E}_{t-1}. \tag{2.54}$$

Secondly, $\mathcal{P}_{t-1}^{\text{sink}} = \mathbb{Q}(\mathcal{V}_{t-1}^{\text{sink}}) \cap \mathcal{E}^{\text{sink}}$. This is depicted in Figure 2.5, and explained in the caption. So,

$$\begin{aligned} I(M; X(\mathcal{P}_{t-1}^{\text{sink}})) &= I(M; X(\mathbb{Q}(\mathcal{V}_{t-1}^{\text{sink}}) \cap \mathcal{E}^{\text{sink}})) && (2.55)\\ &\overset{(a)}{\leq} I(M; X(\mathbb{Q}(\mathcal{V}_{t-1}^{\text{sink}}))) \overset{(b)}{=} 0 && (2.56) \end{aligned}$$

where (a) follows from the fact that considering more random variables can only increase mutual information, and (b) follows from the induction assumption. Finally, consider how

$X(\mathbb{Q}(\mathscr{V}_t^{\text{sink}}))$ depends on M:

$$\begin{aligned} I(M; X(\mathbb{Q}(\mathscr{V}_t^{\text{sink}}))) &= I(M; f_{\mathscr{V}_t^{\text{sink}}}(X(\mathscr{P}_{t-1}^{\text{sink}} \cup \mathscr{P}_{t-1}^{\text{cut}}), W(\mathscr{V}_t^{\text{sink}}))) && (2.57)\\ &\overset{(a)}{\leq} I(M; X(\mathscr{P}_{t-1}^{\text{sink}}), X(\mathscr{P}_{t-1}^{\text{cut}}), W(\mathscr{V}_t^{\text{sink}})) && (2.58)\\ &\overset{(b)}{=} I(M; X(\mathscr{P}_{t-1}^{\text{sink}})) + I(M; X(\mathscr{P}_{t-1}^{\text{cut}}) \,|\, X(\mathscr{P}_{t-1}^{\text{sink}})) && (2.59)\\ &\quad + I(M; W(\mathscr{V}_t^{\text{sink}}) \,|\, X(\mathscr{P}_{t-1}^{\text{sink}}), X(\mathscr{P}_{t-1}^{\text{cut}})) \\ &\overset{(c)}{=} 0, && (2.60) \end{aligned}$$

where once again, (a) and (b) follow from the data processing inequality and the chain rule respectively. In step (c), the first and second terms go to zero by equations (2.56) and (2.54) respectively, while the third term is zero since $W(\mathscr{V}_t^{\text{sink}}) \perp\!\!\!\perp \{M, X(\mathscr{E}_{t-1})\}$ and $\mathscr{P}_{t-1}^{\text{sink}} \cup \mathscr{P}_{t-1}^{\text{cut}} \subseteq \mathscr{E}_{t-1}$.

The proof follows from induction on t, so

$$I(M; X(\mathbb{Q}(\mathscr{V}_t^{\text{sink}}))) = 0 \quad \forall\, t \geq t_0, \tag{2.61}$$

which in turn implies that

$$I(M; X(\mathbb{Q}(V))) = 0 \quad \forall\, V \in \mathscr{V}^{\text{sink}}. \tag{2.62}$$

If there exists an output node whose transmissions depend on M, then there can exist no cut consisting of edges with zero M-information flow, and hence by Lemma 2.8, there must be a path consisting of edges that carry M-information flow between the input nodes and the output node in question. □

2.4.4 The Separability Property

Finally, we state a property that may be of interest to obtain a deeper understanding of the nature of M-information flow, as given by Definitions 2.4 and 2.5.

Proposition 2.9 (Separability)**.** *Let $\mathscr{C}$ be a computational system. Then, at any given point in time t, there exist two sets $\mathscr{R}_t, \mathscr{S}_t \subseteq \mathscr{E}_t$, such that all of the following conditions hold:*

1. $\mathscr{R}_t \cup \mathscr{S}_t = \mathscr{E}_t$
2. $\mathscr{R}_t \cap \mathscr{S}_t = \emptyset$
3. *Either $\mathscr{R}_t = \emptyset$, or for every $R_t \in \mathscr{R}_t$ there exists a subset $\mathscr{R}_t' \subseteq \mathscr{R}_t \setminus \{R_t\}$ such that*
$$I(M; X(R_t) \,|\, X(\mathscr{R}_t')) > 0. \tag{2.63}$$
4. *Either $\mathscr{S}_t = \emptyset$, or for every $\mathscr{E}_t' \subseteq \mathscr{E}_t$,*
$$I(M; X(\mathscr{S}_t) \,|\, X(\mathscr{E}_t')) = 0. \tag{2.64}$$

A proof of this proposition can be found in Appendix 2.B.

Proposition 2.9 shows that at any given point in time t, it is possible to partition $\mathscr{E}_t$ into two sets: $\mathscr{R}_t$, consisting only of edges that have M-information flow, and $\mathscr{S}_t$, comprising edges that have no M-information flow. Furthermore, when considering the M-information flow of edges in $\mathscr{R}_t$, it suffices to condition on the transmissions of edges *within* $\mathscr{R}_t$ to ascertain the presence of M-information flow. Conditioning upon the transmissions of edges in $\mathscr{S}_t$ will not change the mutual information between the message and the transmissions of edges in $\mathscr{R}_t$.

2.5 Inferring Information Flow

Having discussed the definition and the properties of M-information flow, we now consider how these flows of information might be inferred in a real computational system. We first discuss an observation model that describes which random variables are observed and how they are sampled. Under this model, we show how existing techniques from the literature can be used to identify which edges carry M-information flow. As in previous sections, we restrict our attention to detecting *whether or not* a given edge has M-information flow, relegating quantification of these flows to future work. Quantification is briefly discussed in the form of an example in Section 6.5, and again in Section 2.7.4.

We then describe an algorithm that recovers all M-information paths between the input nodes and a given output node, by leveraging the knowledge of which edges have M-information flow. We also explain how one might attain a fine-grained characterization of the structure of information flow, by introducing the concept of "derived information". This is useful for understanding which transmissions are "derived" from others, allowing one to find transmissions that are redundant and discover the presence of hidden nodes. Finally, we explain how flows of information about multiple messages can be inferred in our framework.

2.5.1 The Observation Model

Before we can describe how information flow and information paths can be identified, we must provide a statistical description of the random variables that are observed. Let $\mathscr{C}$ be a computational system under observation. We then make the following assumptions:

1. Transmissions on all edges, including self-edges, are observed. The random variables that are intrinsically generated at each node are *not* observed, unless they are also transmitted on an edge (which could be a self-edge).

2. Several trials[22] are observed, each of which corresponds to an independent realization of all random variables in the model.[23] Every trial uses a realization of M which is

[22] The word "trial" is borrowed from the neuroscience literature, wherein a neuroscientist will often conduct multiple trials in a single experiment. In each trial, a human participant or an animal under study is presented with one of a set of carefully chosen stimuli (corresponding to a realization of the message M in our setting), and neural activity is recorded using some modality. Scientific inferences are then drawn by making use of the activity from all trials.

[23] In reality, trials are not independent in neuroscientific experiments. Indeed, neurons are known to "adapt" their responses from trial to trial, often showing suppressed activity when presented the same stimulus

independently drawn from a distribution determined by the experimentalist.[24] For every node $V \in \mathcal{V}$, the intrinsically generated random variable $W(V)$ is also assumed to be independently and identically distributed across trials.

3. Observations are made noiselessly, in that the realization of each transmission in every trial is observed as-is, without being further corrupted by random noise of any kind. The implications of noisy measurements will be the subject of future work.

Under these conditions, we discuss statistical tests for information flow that are consistent in the asymptotic limit of infinite trials. It should be noted that these assumptions may be valid to varying degrees in different contexts. This is discussed further in Section 7.1.

2.5.2 Detecting Information Flow

Given a sample of all random variables described in the observation model, our next task is to identify which edges have M-information flow. In other words, we need to describe how the conditions given by Definition 2.4 can be rigorously tested, and how we might assert with some confidence that a certain set of edges has information flow at each point in time.

According to Definition 2.4, in order to check whether a particular edge E_t carries M-information flow at time t, we need to test whether at least one of several conditional mutual information quantities is strictly positive. The standard statistical approach for solving this problem is to frame it as a set of "hypothesis tests", which in this case is a set of "conditional independence tests". In general, a hypothesis test formalizes the problem of making an informed decision about the value of some functional of a joint distribution, when observing a sample of data from it. A good conditional independence testing procedure will seek to maximize "statistical power", i.e. the probability of *correctly* identifying the presence of conditional dependence, while keeping the probability of an incorrect identification fixed below some "level" α that is picked beforehand. One intuitive way to do this might be to construct an estimator for the appropriate conditional mutual information, and "reject" the "null" hypothesis of conditional independence if the conditional mutual information was sufficiently larger than some threshold, $\epsilon > 0$. This threshold would have to be chosen so that, on average, the probability of falsely rejecting the null hypothesis is at most α. However, there are usually better ways of performing this test, i.e., it is often possible to attain higher power at the same level *without* actually estimating the conditional mutual information.

While it would be impossible to provide a comprehensive list of papers that have researched the problem of conditional independence testing, it has received (and continues to receive) much attention in the statistics, causality, and information theory communities [99–104]. In its most general form, conditional independence testing is considered to be a hard problem for continuous random variables [105]. However, if we ignore issues associated with the practical difficulty of estimation (discussed later in Section 2.7.1), these works provide

multiple times. This, in part, is considered to be evidence of *learning* in neural circuitry. However, for simplicity, we restrict our attention here to computational systems that do not learn or show trial-to-trial adaptation.

[24] A more detailed discussion of this distribution can be found in Section 2.5.6.

consistent tests under reasonable assumptions on the joint distribution of the variables involved [101–103].

Although we mentioned that there are better ways to test for conditional dependence than to estimate the conditional mutual information, there may be instances when one might want to estimate the conditional mutual information anyway. For instance, in an example that will appear shortly in Section 6.5, we rely on an *estimate* of the conditional mutual information to *quantify* the amount of M-information flowing on a given edge. While our paper has only defined M-information flow in terms of *whether or not* it is present at an edge E_t, it is also extremely useful to know *how much* M-information flow there is. We defer further discussion of this topic until Sections 6.5 and 2.7.4. For now, we note that several papers have considered how to *estimate* mutual information and conditional mutual information, both of which might be essential for an understanding of *quantification* of M-information flow [106–109].

For completeness, we now present a description of how we expect information flow will be detected in practice. We assume that we have a sample of the transmissions from all edges of the computational system, at every point in time. If not, appropriate assumptions may need to be made, as discussed later in Section 7.1. At every instant of time t, consider the set of all edges $\mathscr{E}_t$ present in the network. For every edge $E_t \in \mathscr{E}_t$, use the following process to determine whether it has M-information flow:

1. First test whether the mutual information between its transmission and the message is greater than zero, i.e., $I(M; X(E_t)) > 0$. If so, declare that E_t has M-information flow.

2. If not, test for conditional dependence between its transmission and the message, given each of the other edges E'_t, i.e., test whether $I(M; X(E_t) \,|\, X(E'_t)) > 0$, for each $E'_t \in \mathscr{E}_t \setminus \{\mathscr{E}_t\}$. If any of these tests rejects the null hypothesis, declare that E_t has M-information flow.

3. If not, test for conditional dependence between $X(E_t)$ and M, given subsets of other edges, sequentially conditioning on edges taken pairwise, then in threes, etc. If any of these tests rejects the null, declare that E_t has M-information flow.

4. If none of the above tests rejects the null hypothesis, declare that E_t carries no M-information flow.

Note that we have not discussed the level, α, at which we should reject the null in each of the above tests. In general, since we are performing multiple hypothesis tests simultaneously, some manner of "*correction*" is required to ensure that we do not find, what is effectively, a spurious correlation. This is discussed at length in Section 2.7.1.

2.5.3 Discovering Information Paths

Next, we discuss an algorithm that discovers all M-information paths leading from the input nodes to a given output node, V_{op}, in any computational system. As discussed in Section 2.4, whenever the transmissions $\mathfrak{Q}(V_{\text{op}})$ of the output node depend on the message, Theorem 2.7 guarantees that at least one M-information path exists.

Algorithm 1 Information Path Algorithm: Finds all paths from $\mathscr{V}_{\text{ip}}$ to V_{op}

Initialize an empty graph $\mathscr{H}$ ▷ $\mathscr{H}$ will store valid paths from $\mathscr{V}_{\text{ip}}$ to V_{op}
▷ $\mathscr{H}$ currently contains no nodes or edges
FindInfoPaths($\mathscr{C}$, V_{op}, $\mathscr{H}$) ▷ Call a function (defined below) to populate $\mathscr{H}$
if V_{op} is marked "invalid" **then**
    raise Error ▷ No path from $\mathscr{V}_{\text{ip}}$ to V_{op} was found
end if

function FindInfoPaths($\mathscr{C}$, V_t, $\mathscr{H}$)
    if $\mathscr{P}(V_t)$ is empty **then** ▷ V_t has no inputs $\Rightarrow t = 0$
        if $V_t \in \mathscr{V}_{\text{ip}}$ **then**
            Mark V_t "valid"
            Add V_t to $\mathscr{H}$
        else ▷ We somehow reached a non-input node at $t = 0$
            raise Error
        end if
    else ▷ V_t has inputs
        for all $(U_{t-1}, V_t) \in \mathscr{P}(V_t)$ **do**
            if (U_{t-1}, V_t) has M-information flow **then**
                if U_{t-1} is unmarked **then**
                    FindInfoPaths($\mathscr{C}$, U_{t-1}, $\mathscr{H}$) ▷ This will mark U_{t-1}
                end if
                if U_{t-1} is marked "valid" **then**
                    Mark V_t "valid"
                    Add V_t and (U_{t-1}, V_t) to $\mathscr{H}$
                end if
            end if
        end for
        if V_t is still unmarked **then** ▷ No input of V_t was "valid"
            Mark V_t "invalid"
        end if
    end if
end function

Algorithm 1, which we propose for recovering all M-information paths, is an adaptation of the well-known Depth-First Search[25] method [98, Sec. 22.3]. It takes as its input a computational system $\mathscr{C}$ in which all edges having M-information flow have been identified, the output node V_{op}, and an empty graph $\mathscr{H}$ that is completely devoid of nodes and edges. The algorithm returns the set of all M-information paths in the form of a directed subgraph $\mathscr{H}$ of the time-unrolled graph $\mathscr{G}$. Starting from $\mathscr{V}_{\text{ip}}$, following *any* path in $\mathscr{H}$ will lead one to V_{op}, provided at least one M-information path exists.

[25] It is also possible to discover all M-information paths using an adaptation of Breadth-First Search [98, Sec. 22.2], but doing so would require some mechanism to prune M-information paths that do not lead to the input nodes $\mathscr{V}_{\text{ip}}$. So we prefer to use Depth-First Search for simplicity of exposition.

The algorithm works by recursively visiting nodes, starting from the output node V_{op}. It traverses only edges that carry M-information flow, and uses a marking scheme to avoid revisiting nodes. The same marking scheme is also used to designate nodes to which there are M-information paths from $\mathscr{V}_{\text{ip}}$. As the algorithm passes through each node, it marks the node "valid" whenever an M-information path exists between $\mathscr{V}_{\text{ip}}$ and that node. If no such path exists, then the node is marked "invalid". The objective of the algorithm, therefore, reduces to one of finding a path of "valid" nodes from $\mathscr{V}_{\text{ip}}$ to V_{op}. The algorithm's recursive function can be expressed as follows: *A node $V_t \in \mathscr{V}$ is "valid" if and only if there exists a node $U_{t-1} \in \mathscr{V}$ such that U_{t-1} is valid, and the edge (U_{t-1}, V_t) has M-information flow.* This is a recursive expression since checking the validity of a node at time t involves finding valid nodes at time $t-1$. The only nodes that are considered valid by default are the input nodes $\mathscr{V}_{\text{ip}}$.

The algorithm sequentially checks the validity of nodes $V_t \in \mathscr{V}$, starting from the output node V_{op}. The function FindInfoPaths, when called on any given node V_t, checks the validity of V_t. This involves checking each of the incoming edges of V_t for M-information flow. If U_{t-1} is a node from which M-information flows to V_t, then the algorithm immediately checks the validity of U_{t-1} by calling the function FindInfoPaths again. Eventually, if in this recursive process, we arrive at an input node in $\mathscr{V}_{\text{ip}}$, then that node is marked "valid", and added to the output subgraph $\mathscr{H}$. Once every node U_{t-1} from which M-information flows to V_t has been marked "valid" or "invalid", the validity of V_t can be ascertained. For every "valid" node U_{t-1} from which M-information flows to V_t, the edge (U_{t-1}, V_t) and the node V_t are added to the output subgraph $\mathscr{H}$, and V_t is marked "valid". If there are no such nodes leading to V_t, then V_t is marked "invalid" and does not fall on an M-information path.

This recursive logic yields the set of all M-information paths leading from the input nodes to V_{op}. The two lines at which errors are returned correspond to scenarios that should not occur if the conditions of Theorem 2.7 hold. In line 12, we visit a non-input node at time $t = 0$. But such a node should never have been reached in the recursion, since we only followed edges that have M-information flow. Its presence, therefore, would contradict the computational system model. In line 4, V_{op} is marked "invalid", implying that there is no path leading to it from the input nodes. Once again, this can only occur if the computational system model is violated, or if the conditions of Theorem 2.7 (and Property 2.3) do not hold.

On Computational Complexity

The complexity of Algorithm 1 is exactly that of Depth-first Search, $\mathcal{O}(|\mathscr{V}| + |\mathscr{E}|)$ [98, Sec. 22.3]. To be precise, we consider the computational system to extend until the time of the output node, i.e., we take $T = t_{\text{op}}$. So the complexity of the algorithm is $\mathcal{O}(|\mathscr{V}^*|t_{\text{op}} + |\mathscr{E}^*|t_{\text{op}})$. This is easily verified from the pseudocode listing provided in Algorithm 1: in the worst case, all edges in the system have M-information flow, so all edges and nodes must be traversed by the search. By design of the marking strategy, each node is processed at most once, so we do not need to account for the recursion in any special way. At each node, we must execute lines 7 through 14, and 26 through 28, which take a constant amount of time. Since we have $|\mathscr{V}^*|$ nodes over t_{op} time points, this adds up to $\mathcal{O}(|\mathscr{V}^*|t_{\text{op}})$ steps. We also need to execute the loop in lines 15 through 24, which counts the number of incoming edges at every node. For all nodes combined, this adds up to $\mathcal{O}(|\mathscr{E}^*|t_{\text{op}})$ steps.

If the graph is fully connected as described in Section 2.2, then $|\mathscr{V}^*| = N$ and $|\mathscr{E}^*| = N^2$, so the effective complexity is just $\mathcal{O}(N^2 t_{\text{op}})$. However, if we *know* that the underlying graph is sparse (e.g., because of anatomical priors in neuroscience), then we may have $|\mathscr{E}^*| = \mathcal{O}(N \log N)$, or even $|\mathscr{E}^*| = \mathcal{O}(N)$, bringing down the complexity of the search. It should be noted that in either case, the complexity of identifying *which* edges have M-information flow is potentially exponential in N, as discussed later in Section 2.7.1. This is much larger than the complexity of tracing out information paths, so *finding edges* with M-information flow is, in fact, the "hard part" of the problem.

2.5.4 Derived Information and Redundancy

The framework we develop for information flow allows one to obtain a more fine-grained understanding of information structure in a computational system, especially when compared with classical tools such as correlation and phase synchrony [110, 111]. This allows the experimentalist to better investigate the nature of the computation being performed. A concept that we believe will be extremely useful in this regard is one we call "derived information", which is defined below.

Definition 2.12 (Derived M-Information)**.** *In a computational system $\mathscr{C}$, a transmission $X(Q_t)$ is said to be* derived M-information *of a different transmission $X(P_{t'})$ if M—$X(P_{t'})$—$X(Q_t)$ forms a Markov chain. That is, the following condition must hold:*

$$I(M; X(Q_t) \,|\, X(P_{t'})) = 0, \tag{2.65}$$

implying that

$$H(M \,|\, X(P_{t'})) = H(M \,|\, X(P_{t'}), X(Q_t)). \tag{2.66}$$

So, $X(Q_t)$ adds no new information about M, when given $X(P_{t'})$. The same definition extends to transmissions on sets of edges. Note that, as far as the definition is concerned, t and t' may be any two arbitrary points in time. However, we will typically consider cases when $t \geq t'$.

One potential use-case scenario for derived information arises in the context of redundant flows. Consider the computational system presented in Figure 2.4, originally described under Counterexample 2.3. We see two edges sending the same transmission to the node B_2. This is an example of what we call "redundant transmissions". In general, since we only consider information about M to be relevant, the exact transmissions communicated over two edges at a given point in time may be different. But if they convey the *same information about M* to a given node, then we view them as essentially redundant. Definition 2.4, when applied to this system, will detect both these edges as having M-information flow, since given $X((C_1, B_2))$, their transmissions depend on M. In the notation of the Separability property mentioned earlier (Proposition 2.9), both edges (A_1, B_2) as well as (D_1, B_2) will belong in the set $\mathscr{R}_1$.

Derived information provides a general methodology to understand when transmissions on certain edges may be redundant. Naturally, if the transmissions on two edges Q_t and $P_{t'}$ are redundant, then they must be derived M-information of one another. This amounts to

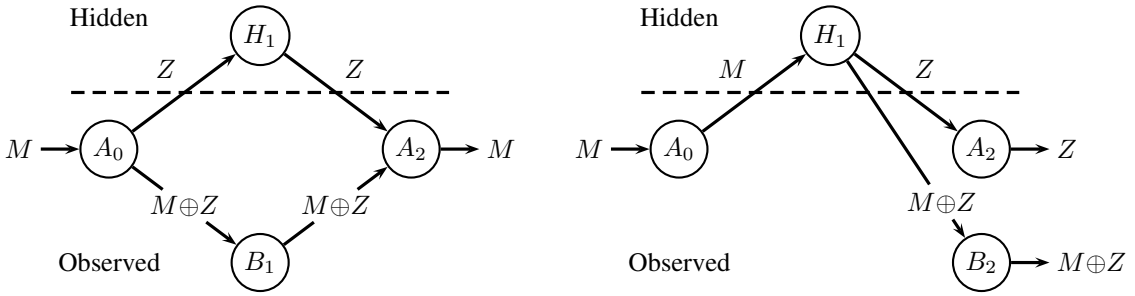


Figure 2.6: Two simple examples showing how hidden nodes may prevent one from being able to discover M-information paths in a computational system. In both cases shown here, H_1 is a hidden node, and we do not observe its incoming or outgoing transmissions. On the left is an example where a transmission that we might need to condition upon to discover M-information flow passes through the hidden node, and therefore cannot be seen. On the right, the hidden node itself generates the source of randomness Z.

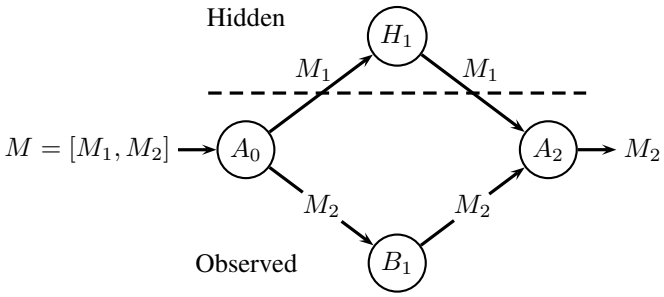


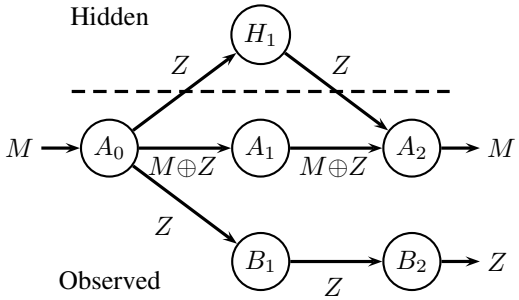
Figure 2.7: A computational system serving as a counterexample to the converse of Proposition 2.11. Here, the hidden node H_1 is M-relevant because its outgoing transmission, M_1, is not present in any of the observed transmissions at time $t = 1$. However, since A_2 chooses to ignore M_1 at its output, the Markov chain M—$X(\widetilde{\mathscr{E}}_1)$—$X(\widetilde{\mathscr{E}}_2)$ boils down to M—M_2—M_2, which obviously holds. Thus, at least based on our current definitions, there may be M-relevant hidden nodes in the system even if Global Markovity continues to hold.

checking two more conditional independence relationships, for which consistent tests exist in the limit of infinite trials, as discussed in Section 2.5.2.

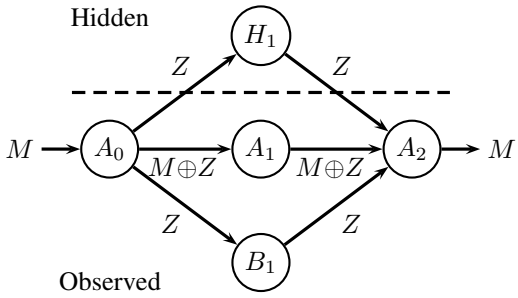
In the following section, we shall see another application of derived information; when applied to specific sets, it can in some cases be used to detect the presence of hidden (unobserved) nodes. Later, in Section 6.5, we discuss an example where the notion of derived information helps us make a new kind of inference about the fine structure of information flow, one that would not be possible using tools such as Granger Causality and Directed Information.

2.5.5 Hidden Nodes

In Section 2.5.3, we showed how the Information Path Algorithm may fail to discover M-information paths if one of the assumptions of the computational system model or the observation model breaks in some way. Here, we discuss one specific situation in which the observation model may break, i.e., when not all nodes are observed. We call these unobserved nodes "hidden nodes", and assume that we do not see transmissions on incoming or outgoing edges of these nodes.



(a) Here, although Global Markovity holds, one could argue that testing for Local Markovity at each node (or at various subsets of nodes) could help uncover the presence of a hidden node.



(b) In this case, the hidden node breaks neither Global nor Local Markovity. However, the *function* computed by A_2 makes *use* of only the hidden node's transmission. As a result, the hidden node has a causal effect on the output of the system, since destroying the outgoing edge of the hidden node would change the output. Such a hidden node is likely undetectable using only observational methods.

Figure 2.8: Examples of computational systems with an M-derived hidden node. In both of these systems, the hidden node's transmission at time $t = 1$ has an effect on the output at A_2. However, Global Markovity continues to hold from $t = 1$ to $t = 2$, because the observed transmissions, $M \oplus Z$ and Z, contain all information necessary to explain the output, M.

Definition 2.13 (Hidden nodes)**.** *Consider a computational system $\mathscr{C} = (\mathscr{G}, X, W, f)$ defined on the time-unrolled graph $\mathscr{G} = (\mathscr{V}, \mathscr{E})$ as before. Suppose that only a subset of nodes in this graph are observed. Specifically, if $\mathscr{V}^*$ was the original set of nodes in $\mathscr{G}^*$, prior to time-unrolling, then we observe only the nodes $\widetilde{\mathscr{V}}^* = \mathscr{V}^* \setminus \mathscr{H}^*$, where $\mathscr{H}^* = \{H^{(0)}, H^{(1)}, \ldots, H^{(K-1)}\}$ is a set of unobserved nodes called* hidden nodes*.*

To describe the observed component of the computational system, we define $\widetilde{\mathscr{E}}^* = \widetilde{\mathscr{V}}^* \times \widetilde{\mathscr{V}}^*$, $\widetilde{\mathscr{V}} = \{V_t : V \in \widetilde{\mathscr{V}}^*, t \in \mathcal{T}\}$ and $\widetilde{\mathscr{E}} = \{(A_t, B_{t+1}) : (A, B) \in \widetilde{\mathscr{E}}^*, t \in \mathcal{T}\}$. Also let $\mathscr{H} = \{H_t : H \in \mathscr{H}^*, t \in \mathcal{T}\}$. Finally, we set up the observed component of the computational system as before: $\widetilde{\mathscr{C}} = (\widetilde{\mathscr{G}}, X, W, f)$. Thus, we only observe the transmissions on edges in $\widetilde{\mathscr{E}}$. As usual, we denote the set of all hidden nodes at time t by $\mathscr{H}_t$, and the set of all observed nodes at time t by $\widetilde{\mathscr{E}}_t$.

The presence of hidden nodes of this nature implies that much of the theory we have developed will not apply. Lemma 2.4 no longer truly holds, in that information about M may persist in the system by passing through the hidden node, even if *no* observed edge has M-information flow. So, naturally, Property 2.1 also fails to hold. Hence, we are not guaranteed to be able to identify all edges with M-information flow, and discover all M-information paths as before. For example, refer to the cases shown in Figure 2.6, where we no can longer find M-information paths because of the presence of a hidden node.

Fortunately, at least in some cases, the concept of derived information (Definition 2.12) provides a simple way to tell whether or not a hidden node exists. Specifically, if at some time t, a hidden node transmits information about M which is unavailable within the system at that time, and which is utilized by some node at the next time instant, then the set

of all observed transmissions $X(\widetilde{\mathscr{E}}_t)$ will *not* be derived M-information of the set of all transmissions at time $t-1$. In other words, the Global Markovity condition (Corollary 2.5) on the observed graph, M—$X(\widetilde{\mathscr{E}}_{t-1})$—$X(\widetilde{\mathscr{E}}_t)$, will break. Unfortunately, the notion of "utilization" is difficult to express mathematically, without resorting to the use of ideas from causality that are based on intervention. The result we prove, therefore, is a simpler sufficiency argument, which guarantees the presence of a hidden node if the aforementioned Markov condition is observed to break. This result is proved in Proposition 2.11, but first, we define some adjectives.

Definition 2.14 (M-relevant hidden node)**.** *A hidden node H_t is said to be* M-relevant *if $\mathbb{Q}(H_t)$ carries M-information flow in $\mathscr{G}$. Similarly, a subset of hidden nodes $\mathscr{H}'_t \subseteq \mathscr{H}_t$ is said to be M-relevant if $\mathbb{Q}(\mathscr{H}'_t)$ carries M-information flow in $\mathscr{G}$.*

Definition 2.15 (M-derived hidden node)**.** *A hidden node H_t is said to be* M-derived *if the Markov chain M—$X(\widetilde{\mathscr{E}}_t)$—$X(\mathbb{Q}(H_t))$ holds. Similarly, a subset of hidden nodes $\mathscr{H}'_t \subseteq \mathscr{H}_t$ is said to be M-derived if the Markov chain M—$X(\widetilde{\mathscr{E}}_t)$—$X(\mathbb{Q}(\mathscr{H}'_t))$ holds.*

Lemma 2.10. *If a subset of hidden nodes is* not *M-derived, then it* is *M-relevant.*[26]

Proposition 2.11. *In a computational system $\mathscr{C}$ with hidden nodes, if Global Markovity on the observed graph, $\widetilde{\mathscr{G}}$, fails to hold from time t to $t+1$, i.e. if $I(M; X(\widetilde{\mathscr{E}}_{t+1}) \,|\, X(\widetilde{\mathscr{E}}_t)) > 0$, then the hidden nodes $\mathscr{H}_t$ at time t are* not *M-derived.*

Proofs of Lemma 2.10 and Proposition 2.11 are straightforward, and are provided in Appendix 2.D. As a direct consequence of these two results, if Global Markovity fails to hold on the observed nodes from time t to $t+1$, then $\mathscr{H}_t$ is M-relevant. By Proposition 2.1, this simply means that there exists at least one M-relevant hidden node at time t.

Although Proposition 2.11 appears to provide a straightforward mechanism to test whether or not hidden nodes exist, it does not always work. If a hidden node's transmissions have no M-information flow, then the node will not be detected. But in this case, it could be argued that such a hidden node does not change whether information paths can be identified, and so can be subsumed by one or more of the intrinsic random variables $W(\cdot)$. Such a hidden node is, therefore, classified by Definition 2.14 as *not* M-relevant.

However, to make matters worse, the converse of Proposition 2.11 does not hold. In particular, there may exist an *M-relevant* hidden node at time t, whose transmission is *ignored* by the node that received it, so that the Markov chain M—$X(\widetilde{\mathscr{E}}_t)$—$X(\widetilde{\mathscr{E}}_{t+1})$ continues to hold (see Figure 2.7). Such a hidden node may *still* be considered largely innocuous.

The most serious case of a hidden node going undetected is one that contains an *M-derived* hidden node, whose transmission *is used* by the receiving node while performing its computation; however the hidden node's transmission is "masked" by a redundant transmission from an observed node (see Figure 2.8). In this case, Global Markovity on $\widetilde{\mathscr{G}}$ will *not* break, yet the hidden node's transmission may be instrumental in producing

[26] If this lemma appears to be somewhat strong, it is only because of the nomenclature "M-derived". For our purposes, a hidden node whose transmissions are independent of the message is also M-derived, since it satisfies the aforementioned Markov condition.

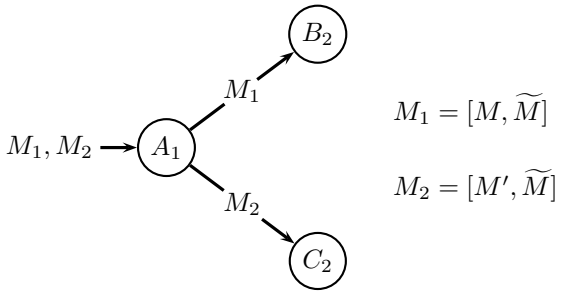


Figure 2.9: A simple example demonstrating the importance of having independent messages (or sub-messages) when exploring the flows of multiple messages in a computational system. As M_1 and M_2 both redundantly contain information about $\widetilde{M}$, both edges shown here have M_1- as well as M_2-information flow. Thus, we are unable to detect the fact that M_1 and M_2 take different paths in the system, because of our choice of stimuli.

a certain output distribution. In some instances, such hidden nodes can be detected by checking for Local Markovity (Proposition 2.6; see Figure 2.8a). However, there are still cases where if we were somehow able to intervene and delete the transmission of the hidden node, then the computational system's output may not remain the same, despite the existence of a redundant transmission from an observed node (see Figure 2.8b). Indeed, the presence of redundancy in such a scenario does not guarantee that the computational system will actually leverage it.

2.5.6 On Multiple Messages and the Distribution of the Message

Just as we can infer information flow and information paths for a single message, we can examine the flows of multiple messages in the same computational system. Consider a case where we wish to understand the information flows of two messages, M_1 and M_2. An neuroscientific example of this might be information flow about two independent components of a visual stimulus, e.g., shape and color (such as in [8]). If $M_1 \perp\!\!\!\perp M_2$, then we could separately identify edges and paths that have M_1-information flow and M_2-information flow, by applying the theory and algorithm as-is for each message individually.

However, if the two messages are *dependent* on one another, one could end up conflating their information flows, based on how they depend on each other, and how the computational system's transmissions carry their joint information. As a simple example, consider the system shown in Figure 2.9, where $M_1 = [M, \widetilde{M}]$ and $M_2 = [M', \widetilde{M}]$, with $M, M', \widetilde{M} \sim$ i.i.d. Ber(1/2). Clearly, M_1 and M_2 both share some redundant information in $\widetilde{M}$, and $I(M_1; M_2) = 1$ bit. Thus, we will see M_1-information flow as well as M_2-information flow on both edges, since the transmission of each edge E satisfies $I(M_i; X(E)) > 0$ for $i \in \{1, 2\}$.

Consider what this means for the aforementioned example of shape and color of a visual stimulus. If a neuroscientist expects that the information paths corresponding to shape and color in the brain are different from each other, what is the best way to design stimuli so as to bring out this difference? Suppose they decided to present a total of four different stimuli, $M \in \{0, 1, 2, 3\}$, with two different shapes and two different colors. Let M_1 be the first bit of the binary representation of M, denoting shape, and M_2 be the second bit, denoting color. Now if the neuroscientist chose to present stimuli with a uniform distribution over M, i.e., if each shape-color combination was shown for one-quarter of all trials, then M_1 and M_2

would be independent of each other, and their individual flows could be tracked separately. However, if the neuroscientist chose to present the four possible stimuli with probabilities $\{1/2, 1/4, 1/8, 1/8\}$ respectively, then M_1 and M_2 are no longer independent of each other, and it may become hard to separate their individual flows as in the example in Figure 2.9.

These examples suggest that, when trying to understand the flows of different messages in a computational system, it helps if they are independent of one another. So from the perspective of experiment design in a neuroscientific context, it is often more sensible to design stimuli so that the two messages of interest are independent of one another. Even when considering a single message that takes one of several values, it becomes important to appropriately choose a distribution over these values to ensure that any sub-messages that are of interest remain independent of one another. This would allow the experimentalist to better understand how "independent dimensions" of the stimulus are processed in the brain.

However, there are also situations where the experimental paradigm necessitates a statistical distribution of stimuli that makes two sub-messages of interest dependent on one another. For instance, the Posner experimental paradigm for attention [112] only works when the proportion of "valid" trials (a certain *type* of trial specific to this paradigm) is roughly 70%. Similarly, during data preprocessing, it is common to discard trials that are excessively noisy, based on some predetermined metric: this process could skew the distribution of the message, even if the original distribution was uniform. If it is still of interest to understand the individual flows of sub-messages in this case, then a possible solution might then be to sub-select experimental trials in such a way as to keep the two sub-messages independent of one another.

2.6 Canonical Computational Examples

In this section, we provide a few canonical examples for computational systems from various contexts. In each case, we discuss what the message M is, and identify which edges carry M-information flow. We also explain how the path recovered by the information path algorithm might be the intuitive choice in each example.

2.6.1 The Butterfly Network from Network Coding

For our first example, we cover the butterfly network from the network coding literature [50, Fig. 7b], reproduced here in Figure 2.10. In this system, we want to communicate two independent bits, $M_1, M_2 \sim$ i.i.d. Ber(1/2), from C_0 to two output nodes, A_4 and B_4. In the network coding context, the butterfly network is the canonical example illustrating that "coding" is necessary to achieve optimal communication: when each edge is restricted to have a capacity of 1 bit, it is not possible to send both M_1 and M_2 simultaneously to A_4 and B_4 using *routing* alone, since we can send only one of M_1 or M_2 on the middle branch (C_2, C_3). We must use *coding*, i.e., we must compute a function of M_1 and M_2, in order to communicate both message bits to A_4 and B_4.

We examine the individual information flows of M_1 and M_2 in the maximal-rate setting where the middle branch carries $M_1 \oplus M_2$. Edges along which information about M_1 flows are colored in blue, while edges along which information about M_2 flows are colored in

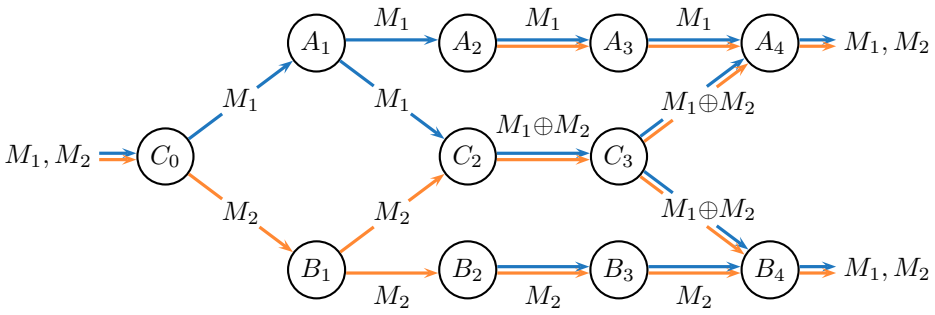


Figure 2.10: A depiction of the butterfly network discussed in Section 2.6.1. There are two messages, M_1 and M_2, each with its own information flow. All edges with M_1-information flow are shown in blue and all edges with M_2-information flow are shown in orange. After time $t = 2$, all edges shown have both M_1- and M_2-information flow. Once the system computes $M_1 \oplus M_2$, edges transmitting M_1 have information flow about both M_1 and M_2, since M_2 can now be decoded from $M_1 \oplus M_2$ and M_1. Furthermore, observe the M_1- and M_2-information paths in this system. In particular, there are two possible M_1-information paths to A_4, but only one possible M_2-information path, which flows through the middle link. The same applies to the M_1-information path to B_4. This may suggest the importance of the middle link in enabling this computation.

orange. The reader may identify these using Definition 2.4 and the transmission on each edge shown in Figure 2.10.

An important feature to observe is that when C_2 mixes information by computing the XOR of M_1 and M_2, we see information about M_1 spontaneously beginning to flow on (B_2, B_3) and similarly, information about M_2 beginning to flow on (A_2, A_3). This is expected, since M_2 is relevant for decoding M_1 at this stage, and indeed, it is exactly this idea which is used to decode M_1 at B_4. All of this is true, despite the fact that $M_1 \oplus M_2$ is independent of M_1 and M_2 individually. This is once again, a prime example of synergy in action.

Applying the information path algorithm (Algorithm 1) for the message M_1 at A_4 will reveal two paths: the "upper path" $(C_0, A_1, A_2, A_3, A_4)$, and the "middle path" $(C_0, A_1, C_2, C_3, A_4)$. However, applying the information path algorithm for the other message M_2 at the same output node A_4 reveals that M_2 exclusively uses the "middle path", $(C_0, B_1, C_2, C_3, A_4)$, to arrive at A_4 from the input nodes.

2.6.2 The Fast Fourier Transform

The Fast Fourier Transform (FFT) is a well-known computational network that provides an intuitive setting for examining information flow. In general, the N-point FFT is an implementation of the N-point Discrete Fourier Transform (DFT), given by

$$\widetilde{Y}_k = \sum_{i=0}^{N-1} Y_i e^{-j\frac{2\pi k}{N}i}, \quad k \in \{0, 1, \ldots, N-1\} \tag{2.67}$$

where j is the imaginary unit. The DFT is a basis transformation of a discrete-time signal Y, which is usually assumed to be periodic with period N. The N-point DFT represents such a signal in the complex-exponential Fourier basis, yielding the Fourier coefficients $\widetilde{Y}$. We consider a simple 4-point DFT, i.e. $N = 4$. The FFT implements this transform using

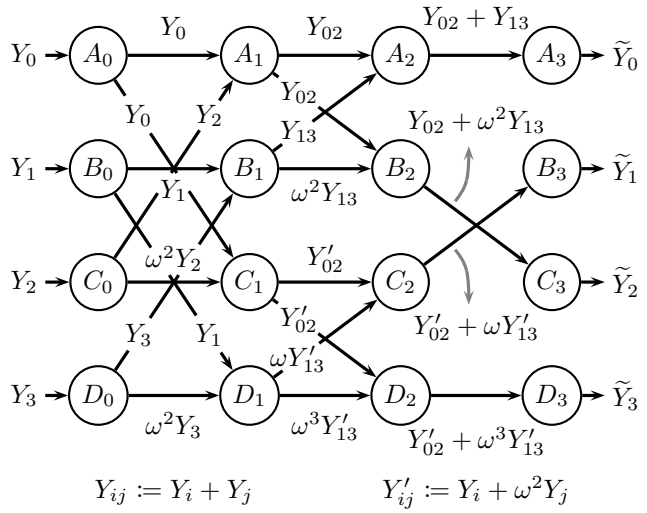


Figure 2.11: The computational system of the 4-point Fast Fourier Transform. For brevity, we have set $\omega \coloneqq e^{-j\frac{2\pi}{4}}$.

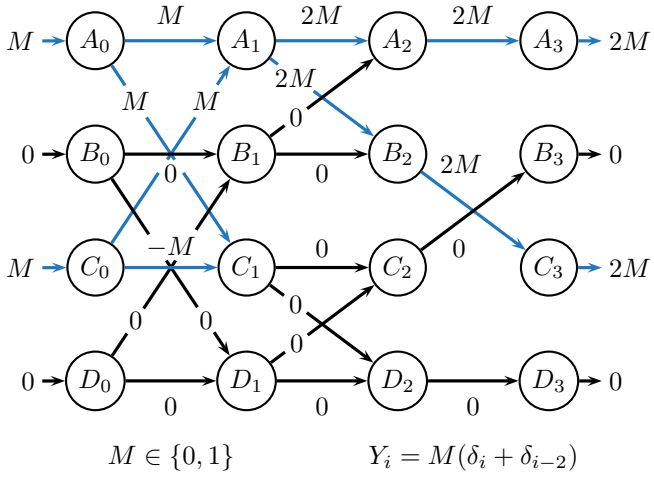


Figure 2.12: An example of information flow in the 4-point FFT, when the message determines which of two signals is supplied to the system: $Y = [0,0,0,0]$ or $Y = [1,0,1,0]$. Observe that, since M is encoded in the even part of Y, only the "even component" of the FFT network is active. Furthermore, only the DC component, $\widetilde{Y}_0$ and the first harmonic, $\widetilde{Y}_2$ are active, as we would expect based on the two input signals.

the computational system shown in Figure 2.11. We refer the reader to [86, Ch. 9] for details. For notational convenience, we have set $\omega = e^{-j\frac{2\pi}{N}} = e^{-j\frac{2\pi}{4}}$.

We use this example to demonstrate how the definition of the message is important in determining information flow. First, suppose the message is one of two signals: $Y = [0,0,0,0]$, or $Y = [1,0,1,0]$. This can be written as $M \in \{0,1\}$ and $Y_i = M(\delta_i + \delta_{i-2})$, where $\delta_i = \mathbb{I}\{i = 0\}$ is the Kronecker Delta function, and we assume $M \sim \text{Ber}(1/2)$. The full computational system, along with the random variables computed on all edges, is shown in Figure 2.12. The edges that have M-information flow are highlighted in blue. Since M is encoded into the *even* part of Y (observe that $Y_i = Y_{-i} \ \forall\, M$), we notice that only the

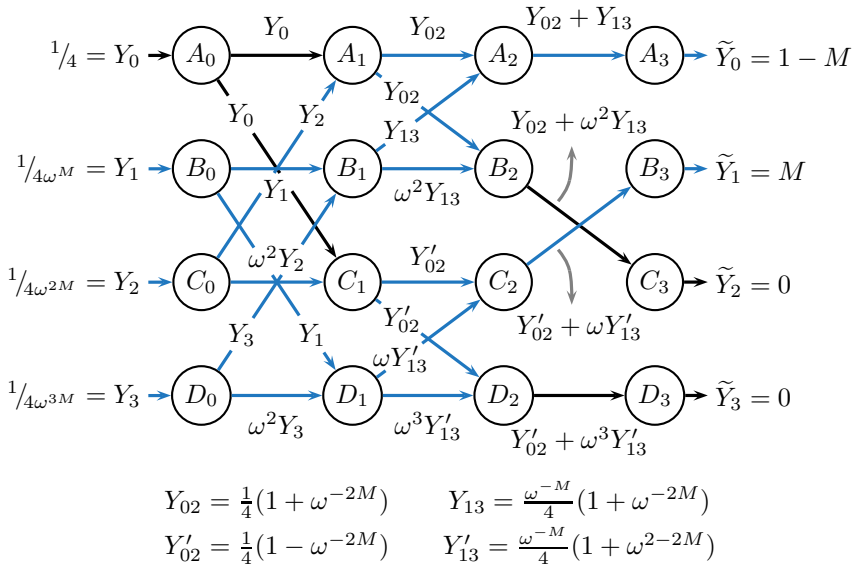


Figure 2.13: Another example of information flow in the 4-point FFT, when the message determines which of two signals is supplied to the system: $Y = [1, 1, 1, 1]$ or $Y = [1, 1/\omega, 1/\omega^2, 1/\omega^3]$. The M-information paths are different from those in Figure 2.12, showing how the choice of the message can have a strong impact on the flows within the same computational system.

"even component" of the FFT system (corresponding to the 2-point FFT on the even indices of Y) is active [86, Sec. 9.3]. Furthermore, only $\widetilde{Y}_0$, the DC component, and $\widetilde{Y}_2$, the first harmonic, show variation with M at the output, as we would expect based on what differs between the two input signals.

As a second example, consider the case shown in Figure 2.13. Here, the message is again one of two signals: $Y = [1, 1, 1, 1]$, or $Y = [1, 1/\omega, 1/\omega^2, 1/\omega^3]$. These signals can be jointly expressed in terms of the binary message random variable $M \sim \text{Ber}(1/2)$ as $Y_i = 1/\omega^{iM}$. The two signals are flat in their magnitude spectra and differ only in their phase, creating δ-functions in the Fourier domain that are frequency-shifted with respect to one another: $\widetilde{Y}_k = \delta_{k-M}$. Once again, the edges in the network that carry M-information flow are demarcated in blue. Refer Appendix 2.F for a derivation of the values of the transmissions in the computational system.

These two examples make it clear that, based on how the message is defined, the M-information paths in the system can be very different. Indeed, if the message were as general as possible, by placing a probability distribution over all possible values of Y in $\mathbb{R}^4$, we know that *all* edges in the computational system would have M-information flow. However, selectively restricting M to just a few signals helps reveal some kind of structure within the FFT network.

Another feature that can be observed in these examples is how the output of the computational system can be a *function* of the message. Although only very simple functions of the message have been shown at the outputs here, the FFT demonstrates that, in principle, more complex functions of the message may also be generated.

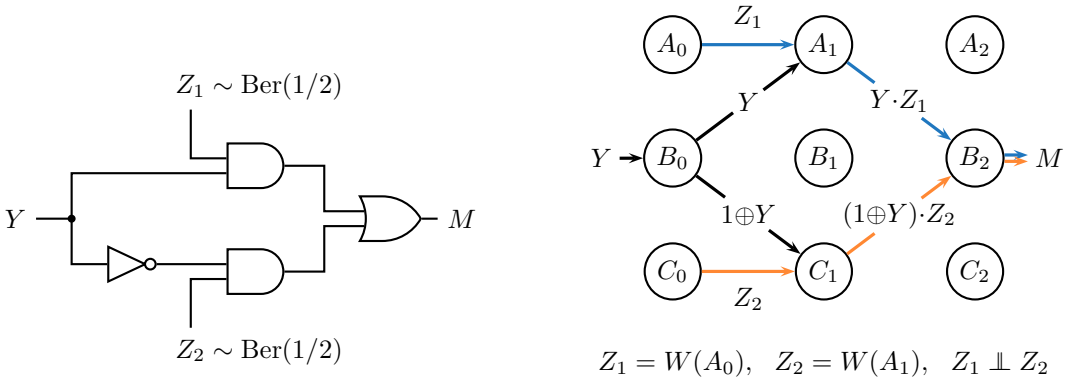


Figure 2.14: A boolean circuit demonstrating a message defined at the output of the computational system. Note that "$\oplus$" refers to bitwise-XOR, "OR" refers to bitwise-OR, and "$\cdot$" refers to bitwise-AND. We see that information paths may lead from an internal node, that generates an intrinsic random variable, to the output node. Furthermore, this path may change with the "external parameters" of the system.

2.6.3 A Message Defined at the Output of a System

We now describe an example where the message is defined at the *output* of a computational system, instead of at the input. Although Definition 2.3c defines the message to be a random variable available at the input nodes, it is also possible to define the message at the output of the computational system. In this scenario, the input nodes are no longer well-defined as per Definition 2.3c. Instead, we would define *output nodes* in the same manner.[27]

Consider the computational system shown in Figure 2.14. The system on the right executes the function depicted by the boolean circuit shown on the left. $Y \in \{0, 1\}$ is an external parameter, which is taken to be a fixed constant. When $Y = 1$, the AND gate at the top is activated while the AND gate at the bottom is deactivated, so the message depends only on Z_1. In this case, only the edges shown in blue have M-information flow. On the other hand, when $Y = 0$, the opposite happens, and the message M depends only on Z_2. Now, only edges shown in orange have M-information flow. If Y was not a deterministic external parameter, but a random variable itself, then all edges shown in the figure would have M-information flow, since M would depend on all their values.

So, we see that when the message is defined at the output, the "origin" of the message may be from within the computation system itself, in the form of one or more intrinsically generated random variables: here, either $Z_1 = W(A_0)$ or $Z_2 = W(C_0)$. The notion of information flow and information paths can thus help us identify where the message originates within the computational system.

Furthermore, just as information paths can change depending upon how the message is defined (as in Section 2.6.2), information paths may also change depending on external

[27] Note, however, that the corresponding "opposite" of Theorem 2.7 does not hold in this case. That is, it is *not true* that if at some previous time instant, an "*input*" node's outgoing transmissions depend on the message, then there exists an information path connecting that input node to the aforementioned output nodes. The reason this fails is that there could be a "source" node at an even earlier time instant, which provides information about M to *both* the input node under consideration, and the output nodes, via two separate, diverging paths. Therefore, there may be no path from said input node to the output nodes.

parameters: inputs such as Y that are fed into the computational system, which are not part of the message. These inputs essentially shape the nature of the computation being performed, and so, naturally, they can affect information paths.

2.7 Discussion

This paper presented a theoretical framework for defining and studying information flow about a specific message in a computational system. The core contribution of our paper was a definition for information flow that is concretely grounded in the *computational task* and intimately tied to a *specific message*. This relied on another important contribution: the development of an underlying computational model, which enables the interpretation of statistical analyses. After providing a clearly-defined model for a computational system, we presented several candidate definitions for information flow along with counterexamples and showed that our definition, which is based on positivity of a conditional mutual information expression, satisfies several intuitive properties, whereas other candidate definitions do not. We then examined these properties in detail and showed, in particular, that our definition naturally leads to the existence of "information paths". We also discussed how information flow can be inferred through conditional independence testing, and provided an algorithm for recovering the information paths in a given system. Finally, we studied some canonical examples of computational systems from different contexts, and showed that our definition of information flow is intuitive in each case.

We proceed to discuss several important assumptions and simplifications in our model. We also discuss existing literature related to estimation of causal influence in neuroscience, and how our computational system model leads us to a significantly different measure of information flow. Similarly, we discuss how our framework is very different from the field of Probabilistic Graphical Models.

2.7.1 The Difficulty of Estimation

A strategy for detecting edges that have M-information flow was presented in Section 2.5.2. In practice, however, there are several issues associated with employing such a strategy. These are discussed below.

Firstly, we currently assume that observations are noiseless (see Section 2.5.1, Assumption 3). It is unclear, exactly, to what extent noisy observations will impact the inference of information flow. In particular, it is worth understanding whether small amounts of observation noise can be tolerated if all edges with M-information flow have a sufficiently large "volume" of information (i.e., the corresponding mutual or conditional mutual information is sufficiently large). As was described intuitively in Section 2.5.2, if the information volume is large, then even under noisy conditions, we might expect the test statistic to clear the threshold, so the presence of M-information flow can still be detected consistently. But small volumes of information that aggregate over time—e.g. information about M "trickling" over time from one node to another—could still pose issues. Such M-information flow could go undetected, as has been shown to occur in other contexts [79], using a different measure of flow. It is possible that Derived Information, in particular, is hard to infer in the presence of

noise. This could make the task of detecting the presence of a hidden node difficult (consider the case of a "trickling" hidden node), as well as that of identifying redundant links.

Secondly, detecting whether each edge at time t has information flow involves checking *all* subsets of $\mathscr{E}_t$. For N nodes and N^2 edges, this implies 2^{N^2} subsets of edges that need to be searched. This could be seen as being prohibitively difficult for $N^2 \geq 30$, or for N greater than about 5 or 6 nodes. However, in reality, graphs in neuroscience are often known to be edge-sparse [113–115]. For example, in the brain, a well-established 11-node network is the reward network [115]. Most nodes in this network typically have just one incoming and one outgoing connection. The two most important nodes have five incoming edges each, with two and four outgoing edges respectively. Further, it is known which connections are inhibitory and which are excitatory, which could further help with testing for information flow. A fully connected network would have had 121 edges, but the underlying connectivity of the circuit only allows for a total of 17 edges in this network. So in reality, anatomical priors help reduce the number of edges to well within the range of what is computable. Nevertheless, it remains of interest to find methods by which nodes and/or edges can be excluded from the search, and this could be another topic for further research.

Another statistical issue that crops up when attempting to simultaneously perform several conditional independence tests is the problem of *multiple comparisons* [116]. Simply put, when performing a large number of independent hypotheses tests, say N, at some fixed false alarm rate α, on average, we should expect αN of these tests to erroneously reject the null. In the context of information flow, we might wish to set the null hypothesis to be the absence of M-information flow on a given edge. Then, to test for M-information flow on this edge, we need to perform a large number of conditional independence tests—call this number N—at some false alarm rate α. These tests are, in fact, *not independent* of one another; nevertheless, very loosely put, if we choose a false alarm rate $\alpha \approx 1/N$, we may find that the probability of *at least one* false alarm is too high. This would make us erroneously infer that this particular edge has M-information flow; moreover, since this argument applies to any edge, if α is not chosen conservatively enough, we may erroneously infer that *all* edges have M-information flow.

This multiple hypothesis testing problem is better posed as a "Global Null test" (e.g., see [117]), wherein the global null is the hypothesis that *all* of the conditional independence tests are individually null (i.e., that there is *no* M-information flow on the given edge), and the global alternative is the hypothesis that *at least one* of the conditional independence tests is non-null (i.e., that there *is* M-information flow on the given edge). As mentioned before, however, the conditional independence tests dictated by Definition 2.4 are, in general, *dependent* on one another. Furthermore, it might not be easy to describe the manner of dependence, so when choosing Global Null tests, it is essential to choose those that work under arbitrary dependence. A simple example of such a test is the well-known Bonferroni correction, which uses a level $\alpha' = \alpha/N$ for each test (where α is the desired false alarm rate for the overall Global Null test); but we may find that such methods have insufficient statistical power. A potential solution to this problem might involve combining multiple Global Null tests in some meaningful way: for example, one could imagine designing a procedure that controls the False Discovery Rate[28] [118] on the *identification of edges*

[28] These methods control the expected proportion of false discoveries, i.e., the proportion of null hypotheses

with M-information flow.[29] Another approach might be to find ways of directly testing information *paths*, wherein the hypothesis tested would be that a certain M-information *path* exists in the system, rather than requiring every *edge* with M-information flow be identified first. All of these ideas are potential avenues for future work.

2.7.2 The Limitations of Granger Causality and Related Tools

Mapping directed functional connectivity and information flow in the brain has been a hot topic for several years, as evidenced by the large body of work in this direction [68–70]. Approaches for statistically mapping functional connectivity often rely on variations of Granger Causality [27] and, more recently, Directed Information [30–32], which we here collectively refer to as "Granger Causality-based tools". These approaches lack a systematic framework that ties the statistical analysis to the underlying computation, however, and the interpretations drawn from their use have often been questioned [72, 74, 75, 79–82].

In particular, a crucial difference between our approach and that of Granger Causality-based tools is that the latter do not have an explicit description of the message. Instead, they provide mechanisms to condense a pair of time series into a single statistic. There are no concrete models that can be used to interpret what this statistic means for the flow of *information* about the message. Furthermore, if one is interested in the information flow of multiple messages, Granger Causality-based tools do not provide an immediate solution. This is why a tool that ties information flow directly with a message is of great interest to practitioners.

The absence of an underlying computational framework with well-defined assumptions inherently makes it very hard to draw sound inferences through the application of Granger Causality-based tools. A striking example of this is a recent result of ours [82] that shows, using a feedback communication system, that the direction of greater Granger-causal influence can be opposite to the direction in which the message is communicated, even in the absence of hidden nodes and measurement noise. The time-unrolled graph framework presented here has been specifically designed to address this issue, and present a clear understanding of information flow, even in the presence of feedback. The example given in Section 6.5 demonstrates a potential resolution to this issue.

Granger Causality was originally developed for the study of time-series that occur only once, such as in economics [26]. An artifact of this development is that it was not designed to incorporate multiple trials of the same process. Instead, it assumes stationarity to help estimate parameters of the random variables that control the process. In the neuroscientific context, stationarity is often a very poor assumption, since the segment of time-series data corresponding to each trial may be short, and often sees some kind of stimulus presentation. Naturally, presentation of the stimulus changes the underlying parameters of the time-series and destroys stationarity; indeed, this is the quintessential aspect of the experiment. Thus, in order to understand processing in such stimulus-driven tasks, one needs to be able to infer time-dependent information flows from data. While information-theoretic extensions of Granger Causality such as Transfer Entropy and Directed Information do not assume

that are falsely rejected.

[29] Care is needed when doing this, however, since tests for M-information flow on different edges at the same time instant are *also* dependent on one another.

stationarity, they nevertheless fail to provide a dynamically evolving picture of information flow. Recently, some work has started to analyze an "adaptive" form of Granger Causality computed on windowed time intervals [84]; such ideas may be worth pursuing for attaining a dynamically evolving picture of Granger causal influence, though they will still not comprise information flow, due to the absence of a connection to the computational task and the message.

In Section 2.1.2, we discussed two dominant interpretations of "information flow" in neuroscience: the first has to do with information about a specific message and is what we address in this paper. The second, having to do with information in the abstract, is more akin to what is done by Granger Causality-based tools. It may be possible, in some settings and under suitable assumptions, to unite these two interpretations. For instance, it would be useful to know under what conditions (e.g., Gaussianity, linear functions, etc.), Granger causal influence provides the same inferences as our rigorous notion of M-information flow. This is a promising future direction, since it is important to understand in which situations Granger Causality-based methods recover meaningful flows of information, and in which cases we must be careful with interpretation.

2.7.3 Probabilistic Graphical Models and Pearl's Causality

There is one important difference that distinguishes our work from the perspective adopted in the field of probabilistic graphical models (PGMs) [119], and the representations therein. In our framework, nodes represent computational units, whereas in PGMs, nodes represent the random variables themselves, and edges capture the conditional independence relationships between these variables. While it might be possible to construct a PGM that is equivalent to our computational model, this would likely eliminate any intuitive structure captured by the computational graph.

It remains to be understood whether and how Pearl's notions of causality [34] can be seamlessly merged with the understanding of information flow developed here. We expect that some formal application of causality will be needed in going from an edge-centric model (as presented here) to a more node-centric one (discussed in Section 7.1), in order to identify which transmissions influenced a given node's output.

There are several works in the literature that discuss measures of information flow in probabilistic graphical models [120, 121], but they are heavily inspired by causality and largely center around an interventionist approach. In contrast, our definition of information flow is based on a computational system model that translates more readily to neuroscience, and we assume that the experimentalist is restricted to making observations. Nevertheless, it may still be interesting to explore alternative definitions of information flow, which incorporate interventional or counterfactual reasoning. Understanding the connection between information flow and interventional approaches could be essential for clinical translation, and constitutes another important direction for future work.

2.7.4 Future Directions for Theoretical Development

A natural question that arises from this paper is: how can our definition of information flow on an edge be extended to a more generic information *measure*, which also quantifies the

volume of flow? Finding such a measure will involve aggregating the conditional mutual information for each subset of edges into a single value (one example of such a measure was provided in Section 6.5, though it was not developed from first-principles). It is as yet unclear how this might be achieved, while still gelling well with our intuition of what this information flow volume ought to be. We believe that the right approach is to start by designating a set of properties that we would like information flow volumes to satisfy, and then to propose a measure through the use of representative examples and counterexamples.

A second direction that emerges is related to Partial Information Decomposition (PID) [89–91], which was discussed earlier in Section 2.3.5. M-information flow is very closely related to the PID: while Candidate Definition 2.1 checks for positivity of mutual information between M and $X(E_t)$, and hence implies the presence of unique and/or redundant information, our definition also detects the presence of purely synergistic information. Since our definition is closely tied to computation and is strongly motivated through the goal of finding unbroken information paths, the close relationship between PID and our definition suggests that PID might be the right toolset for obtaining a more fine-grained understanding of information flow. In particular, it would be useful to know how our understanding of information representation and computation is enhanced through a PID analysis (we try to take this approach in some very preliminary work on error correction in grid cells [122]). Finally, we note that the PID could also help inform the discussion on a definition for information volume. Providing a useful definition of information volume based on current definitions of unique, redundant and synergistic information, and asking whether the problem of information flow can inform the PID literature, will also be the subject of future research.

A third direction has to do with alternate definitions of information flow: there might be other definitions of information about a message, which satisfy the information path property. These could be arrived at through modifications to our current definition, or by looking at directions we did not pursue here, e.g., counterfactual measures. It is worth understanding whether such definitions can avoid M-information orphans, or whether there will be more counterexamples to the use of such measures (we recently made some forays along these lines [123]). Furthermore, the properties we stated in this paper are not sufficient to uniquely specify our definition of information flow. For example, the all-zero function as well as the all-ones function satisfy the Broken Telephone property, although they are not particularly useful definitions of information flow. Thus, it would be useful to understand what other properties we should impose so as to arrive at a unique definition of information flow. As a crude and preliminary example, we demonstrate how this might be done in Appendix 2.E.

2.7.5 Concluding Remarks

We conclude by describing some of our general impressions in working on the theoretical development presented in this paper. As such, these points merely highlight some of our opinions on how theory—and more specifically, information theory—may be applied in neuroscience.

As mentioned in the introduction, we drew inspiration from two papers that discuss how experimentalists understand systems in biology and neuroscience [33, 67]. Both these works advocate for theory by arguing that we need new analytical tools, and that the accumulation of empirical knowledge alone does not constitute *understanding*. Lazebnik [33], in particular,

mentions how terminology in biology tends to be vague and non-committal. We feel that an important reason for this is the absence of concrete underlying models, with clearly-stated assumptions. In other words, we think that theory and modeling can go a long way in providing a *language* that will enable well-grounded discussions. This language, in turn, arises through the development of theoretical models and formal definitions.

Another point made by both the aforementioned papers is that we should attempt to understand large computational systems by first examining smaller models, and models in which the ground truth is already known. This approach allows us to create new analytical tools that can be thoroughly vetted, so that the interpretations drawn from their use in experimental practice is unambiguous and undebated. We also believe that when trying to understand large computational systems, it is essential to start with toy models such as Counterexample 2.1. This philosophy of starting with toy models, and abstracting out meaningful ideas that hold more generally in large systems, is well-entrenched in the field of information theory, and can become a useful export in fields such as neuroscience.

Acknowledgments

We have many people to thank for extremely useful discussions. A non-exhaustive list follows: Mayank Bakshi, Marlene Behrmann, Todd Coleman, Elliot Collins, Uday Jagadisan, Haewon Jeong, Rob Kass, José Moura, Bobak Nazer, Venkatesh Saligrama, Gabe Schamberg, Tsachy Weissman. We also thank the anonymous reviewers whose comments improved our exposition substantially.

Praveen Venkatesh was supported, in part, by a Fellowship in Digital Health from the Center for Machine Learning and Health at Carnegie Mellon University and by a Dowd Fellowship from the College of Engineering at Carnegie Mellon University. The authors would like to thank Philip and Marsha Dowd for their financial support and encouragement. Sanghamitra Dutta was supported, in part, by a K&L Gates Presidential Fellowship in Ethics and Computational Technologies. Pulkit Grover was supported, in part, by an NSF CAREER Award.

2.A Proof of Proposition 2.1

Proof of Proposition 2.1. ($\Rightarrow$) Suppose there exists some edge $E'_t \in \mathscr{E}'_t$ that has M-information flow as per Definition 2.4. That is,

$$\exists\, \mathscr{E}''_t \subseteq \mathscr{E}_t \setminus \{E'_t\} \quad \text{s.t.} \quad I(M; X(E'_t) \,|\, X(\mathscr{E}''_t)) > 0. \tag{2.68}$$

Then,

$$I(M; X(\mathscr{E}'_t) \,|\, X(\mathscr{E}''_t)) = I(M; X(E'_t) \,|\, X(\mathscr{E}''_t)) + I(M; X(\mathscr{E}'_t \setminus \{E'_t\}) \,|\, X(\mathscr{E}''_t), X(E'_t)) \tag{2.69}$$

$$\overset{(a)}{\geq} I(M; X(E'_t) \,|\, X(\mathscr{E}''_t)) \overset{(b)}{>} 0 \tag{2.70}$$

where (a) follows from the non-negativity of conditional mutual information and (b) from (2.68). Taking $\mathscr{R}'_t := \mathscr{E}''_t$ in Definition 2.5, we see that the set $\mathscr{E}'_t$ has M-information flow.

($\Leftarrow$) Next, suppose that the set $\mathscr{E}_t'$ has M-information flow, as per Definition 2.5. That is, there exists a set $\mathscr{R}_t' \subseteq \mathscr{E}_t$ such that

$$I(M; X(\mathscr{E}_t') \,|\, X(\mathscr{R}_t')) > 0. \tag{2.71}$$

Also, let $\{E_t^{(1)}, E_t^{(2)}, \dots E_t^{(K)}\}$ be any ordering of the nodes in $\mathscr{E}_t'$ (where $K = |\mathscr{E}_t'|$). Then by the chain rule of mutual information,

$$\begin{aligned} 0 &< I(M; X(\mathscr{E}_t') \,|\, X(\mathscr{R}_t')) && (2.72)\\ &= \sum_{k=1}^{K} I\Big(M; X(E_t^{(k)}) \,\Big|\, X(\mathscr{R}_t'), X\Big(\bigcup_{j=1}^{k-1} \{E_t^{(j)}\}\Big)\Big). && (2.73) \end{aligned}$$

By the non-negativity of conditional mutual information, at least one of the terms in the summation must be strictly positive. Let the index of this term be k^*. Hence, there exists $E_t' \coloneqq E_t^{(k^*)}$ and $\mathscr{E}_t'' \coloneqq \mathscr{R}_t' \cup \{E_t^{(1)}, \dots E_t^{(k^*-1)}\}$, such that

$$I(M; X(E_t') \,|\, X(\mathscr{E}_t'')) > 0. \tag{2.74}$$

In other words, there exists an edge $E_t' \in \mathscr{E}_t'$ that has M-information flow as per Definition 2.4. □

2.B Proof of Proposition 2.9

Proof of Proposition 2.9. Consider the set of all $E_t \in \mathscr{E}_t$ that have M-information flow. That is, E_t must satisfy

$$\exists\, \mathscr{E}_t' \subseteq \mathscr{E}_t \quad \text{s.t.} \quad I(M; X(E_t) \,|\, X(\mathscr{E}_t')) > 0. \tag{2.75}$$

Define

$$\begin{aligned} \mathscr{R}_t &\coloneqq \{E_t \in \mathscr{E}_t : (2.75) \text{ holds}\},\\ \mathscr{S}_t &\coloneqq \mathscr{E}_t \setminus \mathscr{R}_t. \end{aligned} \tag{2.76}$$

Then, we claim that $\mathscr{R}_t$ and $\mathscr{S}_t$ satisfy equations (2.63) and (2.64).

First, note that if $\mathscr{S}_t \neq \emptyset$, then for every $S_t \in \mathscr{S}_t$, we must have that

$$\forall\, \mathscr{E}_t' \subseteq \mathscr{E}_t, \quad I(M; X(S_t) \,|\, X(\mathscr{E}_t')) = 0. \tag{2.77}$$

If not, then $S_t \in \mathscr{R}_t$ by (2.76), which implies that $S_t \notin \mathscr{S}_t$, which is a contradiction. Hence, we see that no edge in $\mathscr{S}_t$ has M-information flow. Therefore, by Proposition 2.1, the set $\mathscr{S}_t$ has no M-information flow. This directly implies the condition in (2.64).

Next, we claim that if $\mathscr{R}_t \neq \emptyset$, then for every $R_t \in \mathscr{R}_t$, if $\mathscr{E}_t' \subseteq \mathscr{E}_t$ is a set that satisfies

$$I(M; X(R_t) \,|\, X(\mathscr{E}_t')) > 0, \tag{2.78}$$

then $\mathscr{R}_t' \coloneqq \mathscr{E}_t' \cap \mathscr{R}_t$ satisfies

$$I(M; X(R_t) \,|\, X(\mathscr{R}_t')) > 0. \tag{2.79}$$

Let $\mathcal{S}'_t \coloneqq \mathcal{C}'_t \setminus \mathcal{R}'_t$, so that $\mathcal{S}'_t \subseteq \mathcal{S}_t$. Then,

$$I(M; X(R_t) \,|\, X(\mathcal{R}'_t), X(\mathcal{S}'_t)) > 0 \tag{2.80}$$

by (2.78). So,

$$\begin{aligned} I(M; X(R_t) \,|\, X(\mathcal{R}'_t)) &\overset{(a)}{=} I(M; X(R_t), X(\mathcal{S}'_t) \,|\, X(\mathcal{R}'_t)) - I(M; X(\mathcal{S}'_t) \,|\, X(\mathcal{R}'_t), X(R_t)) && (2.81)\\ &\overset{(b)}{=} I(M; X(R_t), X(\mathcal{S}'_t) \,|\, X(\mathcal{R}'_t)) && (2.82)\\ &\overset{(c)}{=} I(M; X(R_t) \,|\, X(\mathcal{R}'_t), X(\mathcal{S}'_t)) + I(M; X(\mathcal{S}'_t) \,|\, X(\mathcal{R}'_t)) && (2.83)\\ &\overset{(d)}{=} I(M; X(R_t) \,|\, X(\mathcal{R}'_t), X(\mathcal{S}'_t)) && (2.84)\\ &\overset{(e)}{>} 0, \end{aligned}$$

where (a) and (c) follow from the chain rule, (b) and (d) follow from (2.64), and (e) follows from (2.80). Thus, condition (2.63) also holds. □

2.C Synergistic Information Flow

2.C.1 Partial Information Decomposition preliminaries

The literature on Partial Information Decomposition seeks to find a decomposition for the mutual information between a message, M, and a set of random variables, $\{X_1, X_2, \ldots\}$ into several individually meaningful, non-negative terms [92]. For our purposes, it suffices to consider the *bivariate* case, i.e., the decomposition of $I(M; X, Y)$ into non-negative components. In the bivariate case, it is well-understood *how many* components there ought to be, and what these quantities *intuitively represent*, but as yet, there is no consensus on a single set of definitions [92].

There is, however, consensus on a basic set of properties that we expect these components to satisfy. For our purposes, we will only make use of the basic properties stated here, so that *any definition* of the aforementioned components which satisfies these properties suffices for our theory.

In the bivariate case, the mutual information between M and (X, Y) is decomposed into four components: information about M which is (i) unique to X and not present in Y, (ii) unique to Y and not present in X, (iii) redundantly present in both X and Y, and (iv) synergistically present in X and Y. In the notation of [91], the decomposition is written as:

$$I(M; (X, Y)) = UI(M : X \setminus Y) + UI(M : Y \setminus X) + RI(M : X; Y) + SI(M : X; Y), \tag{2.85}$$

where the components are ordered exactly as stated above. Note the distinction between the colon and the semicolon in RI and SI. Also, "\" uses the symbol for set-negation to mean "not present in", while also explicitly capturing the asymmetry between X and Y in UI. However, in what follows, we shall assume that RI and SI *are* symmetric in X and Y.

This is usually an additional condition that is imposed when defining these quantities, but here, we take it as given.

Given what we want the four components to represent, we would also expect the following to hold:

$$\begin{aligned} I(M;X) &= UI(M : X \setminus Y) + RI(M : X;Y), \\ I(M;Y) &= UI(M : Y \setminus X) + RI(M : X;Y). \end{aligned} \tag{2.86}$$

As a natural consequence, this means that the conditional mutual information will satisfy:

$$\begin{aligned} I(M;X \,|\, Y) &= I\big(M;(X,Y)\big) - I(M;Y) \\ &= UI(M : X \setminus Y) + SI(M : X;Y), \\ I(M;Y \,|\, X) &= I\big(M;(Y,X)\big) - I(M;X) \\ &= UI(M : Y \setminus X) + SI(M : X;Y). \end{aligned} \tag{2.87}$$

Finally, we want each of these components to always be non-negative:

$$\begin{aligned} UI(M : X \setminus Y) &\geq 0 \qquad & RI(M : X;Y) &\geq 0 \\ UI(M : Y \setminus X) &\geq 0 \qquad & SI(M : X;Y) &\geq 0. \end{aligned} \tag{2.88}$$

It is not obvious that a consistent definition of these four quantities which also satisfies the equations stated above even *exists*, but in fact, additional properties are required to obtain a unique definition. For instance, see [91] for one such development.

As stated before, our theory only relies on the properties stated in this section. As a result, our theorem on the equivalence of information flow definitions holds irrespective of what definition is used, exactly, for synergistic information. It only matters that the definition used satisfies the basic properties presented here.

2.C.2 Equivalence of information flow definitions

Proof of Proposition 2.2. ($\Rightarrow$) Suppose the edge E_t has M-information flow. Then,

$$\exists\, \mathscr{C}_t' \subseteq \mathscr{C}_t \quad \text{s.t.} \quad I\big(M; X(E_t) \,|\, X(\mathscr{C}_t')\big) > 0. \tag{2.89}$$

If $I\big(M;X(E_t)\big) > 0$ with $\mathscr{C}_t' = \emptyset$ in (2.89), then condition 1 in Definition 2.6 holds, so nothing remains to be shown. If not, then $I\big(M;X(E_t)\big) = 0$, so (2.89) implies that there must exist some $\mathscr{C}_t' \neq \emptyset$ such that

$$I\big(M; X(E_t) \,|\, X(\mathscr{C}_t')\big) > 0, \tag{2.90}$$

which, by (2.87), is equivalent to

$$UI\big(M : X(E_t) \setminus X(\mathscr{C}_t')\big) + SI\big(M : X(E_t); X(\mathscr{C}_t')\big) > 0. \tag{2.91}$$

However, since $I\big(M;X(E_t')\big) = 0$, we must have $UI\big(M : X(E_t) \setminus X(\mathscr{C}_t')\big) = 0$ by (2.86) and (2.88). Hence,

$$\exists\, \mathscr{C}_t' \subseteq \mathscr{C}_t \setminus \{E_t\} \quad \text{s.t.} \quad SI\big(M : X(E_t); X(\mathscr{C}_t')\big) > 0. \tag{2.92}$$

So the implication in the forward direction holds.

($\Leftarrow$) For the converse, suppose that E_t has no M-information flow. That is,

$$I(M; X(E_t) \,|\, X(\mathscr{C}'_t)) = 0 \quad \forall\, \mathscr{C}'_t \subseteq \mathscr{C}_t \setminus \{\mathscr{C}_t\}. \tag{2.93}$$

By (2.87), this implies that

$$UI(M : X(E_t) \setminus X(\mathscr{C}'_t)) + SI(M : X(E_t); X(\mathscr{C}'_t)) = 0 \quad \forall\, \mathscr{C}'_t \subseteq \mathscr{C}_t \setminus \{\mathscr{C}_t\}. \tag{2.94}$$

Since UI and SI are both non-negative by (2.88), we must have that

$$SI(M : X(E_t); X(\mathscr{C}'_t)) = 0 \quad \forall\, \mathscr{C}'_t \subseteq \mathscr{C}_t \setminus \{\mathscr{C}_t\}. \tag{2.95}$$

This proves the converse. □

2.D Miscellaneous Proofs from Section 2.5

2.D.1 Proof of Lemma 2.10

Proof of Lemma 2.10. Consider a subset of hidden nodes $\mathscr{H}'_t \subseteq \mathscr{H}_t$ that is not M-relevant. Then, by Definition 2.14, $\mathbb{Q}(\mathscr{H}'_t)$ carries no M-information flow in $\mathscr{G}$. This means that

$$\forall\, \mathscr{C}'_t \subseteq \mathscr{C}_t, \quad I(M; X(\mathbb{Q}(\mathscr{H}'_t)) \,|\, X(\mathscr{C}'_t)) = 0. \tag{2.96}$$

Specifically, taking $\mathscr{C}'_t = \widetilde{\mathscr{C}}_t$, we have

$$I(M; X(\mathbb{Q}(\mathscr{H}'_t)) \,|\, X(\widetilde{\mathscr{C}}_t)) = 0. \tag{2.97}$$

Therefore, by Definition 2.15, $\mathscr{H}'_t$ is M-derived. Thus, if $\mathscr{H}'_t$ is *not* M-relevant, it *is* M-derived. Taking the contrapositive, if $\mathscr{H}_t$ is *not* M-derived, then it *is* M-relevant. □

2.D.2 Proof of proposition 2.11

Proof of Proposition 2.11. We are given that

$$I(M; X(\widetilde{\mathscr{C}}_{t+1}) \,|\, X(\widetilde{\mathscr{C}}_t)) > 0, \tag{2.98}$$

and must prove that the hidden nodes at time t, $\mathscr{H}_t$, are *not* M-derived.

First note that, since $\mathbb{Q}(\widetilde{\mathscr{V}}_{t+1}) = \widetilde{\mathscr{C}}_{t+1} \cup (\widetilde{\mathscr{V}}_{t+1} \times \mathscr{H}_{t+2})$, we must have

$$\begin{aligned}
&I(M; X(\mathbb{Q}(\widetilde{\mathscr{V}}_{t+1})) \,|\, X(\widetilde{\mathscr{C}}_t)) \\
&\quad = I(M; X(\widetilde{\mathscr{C}}_{t+1}), X(\widetilde{\mathscr{V}}_{t+1} \times \mathscr{H}_{t+2}) \,|\, X(\widetilde{\mathscr{C}}_t)) && (2.99)\\
&\quad = I(M; X(\widetilde{\mathscr{C}}_{t+1}) \,|\, X(\widetilde{\mathscr{C}}_t)) + I(M; X(\widetilde{\mathscr{V}}_{t+1} \times \mathscr{H}_{t+2}) \,|\, X(\widetilde{\mathscr{C}}_{t+1}), X(\widetilde{\mathscr{C}}_t)) && (2.100)\\
&\quad \geq I(M; X(\widetilde{\mathscr{C}}_{t+1}) \,|\, X(\widetilde{\mathscr{C}}_t)) && (2.101)\\
&\quad > 0, && (2.102)
\end{aligned}$$

where the last line follows from the fact that conditional mutual information is non-negative, and from (2.98).

Next, observe that Local Markovity conditions (Proposition 2.6) *must* hold on the *entire* graph $\mathcal{G}$, which consists of both observed and hidden nodes. If we apply the Local Markovity condition to $\widetilde{\mathcal{V}_{t+1}}$, we have M—$X(\mathcal{P}(\widetilde{\mathcal{V}_{t+1}}))$—$X(\mathbb{Q}(\widetilde{\mathcal{V}_{t+1}}))$, or in other words

$$I\big(M; X(\mathbb{Q}(\widetilde{\mathcal{V}_{t+1}})) \,\big|\, X(\mathcal{P}(\widetilde{\mathcal{V}_{t+1}}))\big) = 0. \tag{2.103}$$

Note that $\mathcal{P}(\widetilde{\mathcal{V}_{t+1}}) = \widetilde{\mathcal{E}}_t \cup \widetilde{\mathbb{Q}}(\mathcal{H}_t)$, where $\widetilde{\mathbb{Q}}(\mathcal{H}_t) \coloneqq \mathcal{H}_t \times \widetilde{\mathcal{V}_{t+1}}$ is the subset comprising outgoing edges of $\mathcal{H}_t$ that go to $\widetilde{\mathcal{V}_{t+1}}$. Therefore,

$$I\big(M; X(\mathbb{Q}(\widetilde{\mathcal{V}_{t+1}})) \,\big|\, X(\widetilde{\mathcal{E}}_t), X(\widetilde{\mathbb{Q}}(\mathcal{H}_t))\big) = 0. \tag{2.104}$$

Expanding this conditional mutual information, we get

$$I\big(M; X(\mathbb{Q}(\widetilde{\mathcal{V}_{t+1}})), X(\widetilde{\mathbb{Q}}(\mathcal{H}_t)) \,\big|\, X(\widetilde{\mathcal{E}}_t)\big) - I\big(M; X(\widetilde{\mathbb{Q}}(\mathcal{H}_t)) \,\big|\, X(\widetilde{\mathcal{E}}_t)\big) = 0. \tag{2.105}$$

So we have

$$\begin{aligned}
&I\big(M; X(\widetilde{\mathbb{Q}}(\mathcal{H}_t)) \,\big|\, X(\widetilde{\mathcal{E}}_t)\big) \\
&\quad = I\big(M; X(\mathbb{Q}(\widetilde{\mathcal{V}_{t+1}})), X(\widetilde{\mathbb{Q}}(\mathcal{H}_t)) \,\big|\, X(\widetilde{\mathcal{E}}_t)\big) && (2.106)\\
&\quad = I\big(M; X(\mathbb{Q}(\widetilde{\mathcal{V}_{t+1}})) \,\big|\, X(\widetilde{\mathcal{E}}_t)\big) + I\big(M; X(\widetilde{\mathbb{Q}}(\mathcal{H}_t)) \,\big|\, X(\mathbb{Q}(\widetilde{\mathcal{V}_{t+1}})), X(\widetilde{\mathcal{E}}_t)\big) > 0, && (2.107)
\end{aligned}$$

where the final inequality follows from (2.102) and the fact that conditional mutual information is non-negative. Finally, since $\widetilde{\mathbb{Q}}(\mathcal{H}_t) \subset \mathbb{Q}(\mathcal{H}_t)$, we have that $I\big(M; X(\mathbb{Q}(\mathcal{H}_t)) \,\big|\, X(\widetilde{\mathcal{E}}_t)\big) > 0$, just as we showed in equations (2.99)–(2.102). Hence, the Markov chain M—$X(\widetilde{\mathcal{E}}_t)$—$X(\mathbb{Q}(\mathcal{H}_t))$ does not hold, so by Definition 2.15, $\mathcal{H}_t$ are not M-derived. □

2.E On the Uniqueness of Our Definition of Information Flow

From the perspective of designing an axiomatic framework, it is desirable to find a minimal set of properties that gives rise to a unique definition of information flow. Although Property 2.1 helped us motivate a definition for information flow, it did not uniquely specify a definition. Indeed, the all-zero function as well as the all-ones function also satisfy the property, although they are not particularly useful definitions of information flow.

In this section, we provide a set of properties that uniquely leads to our definition of information flow. However, we must acknowledge that we arrived at these properties with the benefit of hindsight, after having proved many other properties of our definition. As such, they are mathematically very similar to our definition, and one might feel uncomfortable with the idea of imposing such a set of properties at the very outset. Our goal here is only to begin a discussion in this direction: a search for a more abstract set of properties that leads to a unique definition of information flow would be a worthy endeavour in future.

Property 2.4. *Let $\mathscr{C}$ be a computational system, and let $\mathscr{F}_M : \mathscr{C} \to \{0,1\}$ be an indicator of the presence of information flow about M on an edge. That is, $\mathscr{F}_M(E) = 1$, if information about M flows on the edge $E \in \mathscr{C}$, and $\mathscr{F}_M(E) = 0$ otherwise. We now state three conditions $\mathscr{F}_M$ must satisfy, which naturally leads to our definition of information flow (Definition 2.4):*

2.4a) $\mathscr{F}_M(E_t) = 1$ *if* $I(M; X(E_t)) > 0$

2.4b) $\mathscr{F}_M(E_t) = 1$ *if* $\exists\, \mathscr{C}_t' \subseteq \mathscr{C}_t \backslash \{E_t\}$ *s.t.*

$$I(M; X(\mathscr{C}_t') \,|\, X(E_t)) > I(M; X(\mathscr{C}_t'))$$

2.4c) $\mathscr{F}_M(E_t) = 0$ *if* $I(M; X(E_t) \,|\, X(\mathscr{C}_t')) = 0 \;\forall\, \mathscr{C}_t' \subseteq \mathscr{C}_t$.

Property 2.4a is a very natural and intuitive requirement for information flow. Property 2.4b states that an edge should be considered to carry information about M, if upon conditioning, its transmission *increases* the information that some set $X(\mathscr{C}_t')$ conveys about M. Property 2.4c is reminiscent of the separability property from Proposition 2.9, and states that if an edge has no dependence with M, no matter what other transmission is conditioned upon, then it can carry no information flow about M.

Effectively, Property 2.4a states that if an edge has unique or redundant information about M, then it must carry information flow, while Property 2.4b states that if an edge has synergistic information about M along with some other set of transmissions, then it must carry information flow. Finally, Property 2.4c states that if all three of these components are absent, then that edge carries no information flow. This also explains how, if any one of these three properties is absent, our definition is no longer unique.

As we acknowledged previously, some of these properties could be seen as too restrictive or contrived, and a more abstract set of properties is certainly desirable. Nevertheless, these properties do uniquely identify our definition of information flow.

Proposition 2.12 (Uniqueness)**.** *If $\mathscr{F}_M$ is an indicator of information flow that satisfies the conditions in Property 2.4, then $\mathscr{F}_M(E_t) = 1$ if and only if E_t has M-information flow, per Definition 2.4.*

Proof. ($\Rightarrow$) Suppose the edge E_t has no M-information flow per Definition 2.4. This directly implies the condition in Property 2.4c. Hence, $\mathscr{F}_M(E_t) = 0$. This proves that if $\mathscr{F}_M(E_t) = 1$, the edge E_t must have M-information flow.

($\Leftarrow$) Suppose the edge E_t *has* M-information flow per Definition 2.4. Then,

$$\exists\, \mathscr{C}_t' \subseteq \mathscr{C}_t \backslash \{E_t\} \quad \text{s.t} \quad I(M; X(E_t) \,|\, X(\mathscr{C}_t')) > 0. \tag{2.108}$$

If $\mathscr{C}_t' = \emptyset$, $I(M; X(E_t)) > 0$, so by Property 2.4a, $\mathscr{F}_M(E_t) = 1$. If $I(M; X(E_t)) = 0$, then (2.108) guarantees the existence of some $\mathscr{C}_t' \neq \emptyset$ such that

$$I(M; X(E_t) \,|\, X(\mathscr{C}_t')) > 0 \tag{2.109}$$

$$\Rightarrow \quad I(M; X(\mathscr{C}_t')) + I(M; X(E_t) \,|\, X(\mathscr{C}_t')) \overset{(a)}{>} I(M; X(\mathscr{C}_t')) \tag{2.110}$$

$$\Rightarrow \quad I(M; X(E_t), X(\mathscr{C}_t')) \overset{(b)}{>} I(M; X(\mathscr{C}_t')) \tag{2.111}$$

$$\Rightarrow \quad I(M;X(E_t)) + I(M;X(\mathscr{C}'_t)\,|\,X(E_t)) \overset{(c)}{>} I(M;X(\mathscr{C}'_t)) \tag{2.112}$$

$$\Rightarrow \quad I(M;X(\mathscr{C}'_t)\,|\,X(E_t)) \overset{(d)}{>} I(M;X(\mathscr{C}'_t)), \tag{2.113}$$

where in (a), we simply added $I(M;X(\mathscr{C}'_t))$ to both sides; in (b) and (c), we used the chain rule in two different ways; and in (d), we used the fact that $I(M;X(E_t)) = 0$. So, by Property 2.4b, we have that $\mathscr{F}_M(E_t) = 1$. This proves the converse. □

Remark It should be noted that Definition 2.4 only specifies *whether or not* a given edge has M-information flow. It does not *quantify* this flow. So Proposition 2.12 demonstrates the uniqueness of our definition up to an unspecified information volume. If we require that the conditions in Property 2.4 hold, then any quantitative definition of information flow will go to zero at an edge if and only if the M-information flow carried by that edge is zero.

2.F Derivation of Expressions in the Second FFT Example from Section 2.6.2

Here, we derive the expressions used in Figure 2.13. Recall that $Y_i = \omega^{-iM}/4$, where $\omega = e^{-j2\pi/4} = -j$.

$$Y_{02} = Y_0 + Y_2 = \frac{1}{4} + \frac{\omega^{-2M}}{4} = \frac{1}{4}(1+\omega^{-2M}) \tag{2.114}$$

$$Y_{13} = Y_1 + Y_3 = \frac{\omega^{-M}}{4} + \frac{\omega^{-3M}}{4} = \frac{\omega^{-M}}{4}(1+\omega^{-2M}) \tag{2.115}$$

$$Y'_{02} = Y_0 + \omega^2 Y_2 = \frac{1}{4} + (-1)\frac{\omega^{-2M}}{4} = \frac{1}{4}(1-\omega^{-2M}) \tag{2.116}$$

$$Y'_{13} = Y_1 + \omega^2 Y_3 = \frac{\omega^{-M}}{4} + (-1)\frac{\omega^{-3M}}{4} = \frac{\omega^{-M}}{4}(1-\omega^{-2M}) \tag{2.117}$$

Next, we show that these intermediate values actually yield the expected values of $\widetilde{Y}$.

$$\widetilde{Y}_0 = Y_{02} + Y_{13} = \frac{1}{4}(1+\omega^{-2M}+\omega^{-M}+\omega^{-3M}) \tag{2.118}$$

$$= \begin{cases} \frac{1}{4}(1+1+1+1), & M=0 \\ \frac{1}{4}(1+j+j^2+j^3), & M=1 \end{cases} \tag{2.119}$$

$$= 1 - M \tag{2.120}$$

$$\widetilde{Y}_1 = Y'_{02} + \omega Y'_{13} = \frac{1}{4}(1-\omega^{-2M}+\omega^{1-M}-\omega^{1-3M}) \tag{2.121}$$

$$= \begin{cases} \frac{1}{4}(1-1+\omega-\omega), & M=0 \\ \frac{1}{4}(1-j^2+1-j^2), & M=1 \end{cases} \tag{2.122}$$

$$= M \tag{2.123}$$

$$\widetilde{Y}_2 = Y_{02} + \omega^2 Y_{13} = \frac{1}{4}(1+\omega^{-2M}+\omega^{2-M}+\omega^{2-3M}) \tag{2.124}$$

$$= \frac{1}{4}(1 + \omega^{-2M} - \omega^{-M} - \omega^{-3M}) \tag{2.125}$$

$$= \begin{cases} \frac{1}{4}(1 + 1 - 1 - 1), & M = 0 \\ \frac{1}{4}(1 - 1 - \omega^{-1} + \omega^{-1}), & M = 1 \end{cases} \tag{2.126}$$

$$= 0 \tag{2.127}$$

$$\widetilde{Y}_3 = Y'_{02} + \omega^3 Y'_{13} = \frac{1}{4}(1 - \omega^{-2M} + \omega^{3-M} - \omega^{3-3M}) \tag{2.128}$$

$$= \frac{1}{4}(1 - \omega^{-2M} + \omega^3(\omega^{-M} - \omega^{-3M})) \tag{2.129}$$

$$= \begin{cases} \frac{1}{4}(1 - 1 - \omega(1 - 1)), & M = 0 \\ \frac{1}{4}(1 - (-1) - \omega(\omega^{-1} + \omega^{-1})), & M = 1 \end{cases} \tag{2.130}$$

$$= 0 \tag{2.131}$$

3 M-Information Flow in Neuroscience

Try not, do. Or do not.
There is no try.
— Master Yoda

3.1 Introduction

This chapter describes how M-information flow might be useful in a neuroscientific context through simulations on biological neural network models. The chapter considers two important ways in which the M-information flow framework can influence neuroscientific experiments:

1. First, the framework communicates the importance of accounting for synergy when tracking information flows about a message. This provides an additional justification for the use of measures from the partial information decomposition literature in neuroscientific data analysis. To demonstrate this fact, we present simulations on networks of reparametrized quadratic-integrate-and-fire neurons, which are capable of generating and maintaining synergistic representations. We also propose measures of information flow that capture the essence of our definition in Chapter 2, but which are based on correlation and are hence much easier to compute.

2. Second, we discuss how the use of partial information measures, such as uniqueness, redundancy and synergy, can suggest new hypotheses about information representation. Since partial information measures are essential for information flow, we consider how they can be useful in their own right, for understanding encoding. We examine a simple model of entorhinal grid cells, and show how unique, redundant and synergistic information can all arise in this model when using different encoding schemes to represent information about spatial location.

The chapter uses the unifying theme of synergistic information: how it is important for tracking flows of information, and how it can arise in surprising ways in encoding schemes for grid cells.

3.2 A Synergistic Perspective on Information Flow and Encoding

Synergy informally refers to the notion that a whole can be more than the sum of its parts. In information theory, this takes the form of two variables X and Y *jointly* conveying some information about a message M that cannot be obtained from either one of them *individually*. While this statement offers only a vague intuition for synergy, several recent works in the information theory literature have proposed concrete definitions for synergy [89–91] (see Lizier et al. [92] for a recent review). This field, called Partial Information Decomposition (PID), provides formal definitions for unique, redundant and synergistic information. Some of these definitions are rooted in strong operational interpretations, relying on ideas from statistical decision theory [91, 124]. There have also been significant efforts towards finding efficient and practical estimators for these definitions [125]. These advances suggest that partial information measures—measures of unique, redundant and synergistic information—are ready to be used in neuroscience.

Synergy has been explored by many works in neuroscience over the last two and a half decades. For instance, Schneidman et al. [93] identified three different kinds of "independence" in the neural code—activity independence, conditional independence and information independence—the last of these is related to the synergy between different cells about the stimulus. Gat and Tishby [96] experimentally identified the presence of synergy between neurons in the frontal or prefrontal cortex of monkeys performing a Go/No-go task. More recently, works by Timme and Lapish [95] and Pica et al. [126] have described how partial information measures such as unique, redundant and synergistic information may be used in neuroscience. We revisit the works of both Schneidman et al. [93] and Timme and Lapish [95] in a later section, providing additional context and contrasting some of the finer points of our results with theirs.

Despite the existence of past work addressing synergy in a multitude of ways, we believe that this concept deserves to be re-examined. Our argument rests on two points, both of which we illustrate in this paper using simulations:

1. A recent result of ours [49] theoretically argues that one *must* account for synergy in some form, in order to provably *track* how information about a stimulus flows through the brain. Understanding such dynamical flows of information in the brain, in turn, is essential if we wish to intervene to modify these flows, particularly for the treatment of various brain diseases and disorders. Unless we account for synergy, there will always be instances where we cannot consistently identify the flow of information about a stimulus or response. In the present work, we provide concrete examples of such instances by simulating circuits at three different scales of neural processing.

2. We also show that synergy can arise in the brain in surprising ways, using a case study on grid cells. Borrowing from existing models of encoding and error correction in grid cells [127], we build a model for the joint activity of three grid cell modules and examine how information about spatial location is decomposed between these modules. These simulations show that when interrogating information about spatially refined location, each module provides unique information with respect to the others; and when these grid modules possess the capacity for error correction, they provide redundant

information with respect to the others. However, when interrogating information about location at a coarse spatial resolution, grid cells encode the same information synergistically.

These two results illustrate that synergy may be more common than previously assumed, and even unexpected in some instances, while at the same time being essential for understanding information flows. In conjunction with the aforementioned advances in developing well-motivated measures of synergistic information, questions examining synergy in the brain appear to be ripe for further investigation through experimental studies.

The main results of this paper are organized into three sections. The first deals with the requisite background knowledge for understanding synergy and the associated partial information decomposition literature. We then present the results of our simulations showing the connection between synergy and information flow: this is presented in the form of three experiments, followed by a few general remarks. The last section of our results shows how synergy may arise in entorhinal grid cells in ways perhaps unexpected. We then present the details of our simulations in the methods section, and finally conclude with a discussion on the key takeaways from our work.

3.3 The Partial Information Decomposition Framework

Before we can explore the importance of synergy, we first provide an intuitive explanation of what synergy *means*. Then, we describe a few different ways in which prior literature in neuroscience has tried to understand synergy. Finally, we describe how synergy has been formalized through recent advances in the information theory literature. This section assumes that the reader is familiar with basic information-theoretic concepts such as Shannon entropy and mutual information [128]. For convenience, these are summarized in Table 3.1 at the end of this section.

Intuitively, synergy refers to the idea two variables X and Y can provide *more* information about some message M when taken *together*, than when considered *individually*. Schneidman et al. [93] operationalized this intuition literally, by considering the difference of total and individual mutual informations. They defined a quantity that we denote $\text{Syn}(M : X; Y)$, referring to the aforementioned difference:

$$\text{Syn}(M : X; Y) \coloneqq I(M; (X, Y)) - I(M; X) - I(M; Y) \tag{3.1}$$

Schneidman et al. [93] argued (correctly) that if this quantity was positive, then X and Y had synergistic information about M, and that if it was negative, then they had redundant information about M.

However, as we shall see, this definition suffered from an important issue, i.e., it did not allow for both synergy and redundancy to be present simultaneously. In fact, the aforementioned quantity was the difference of synergistic and redundant information: thus, while $\text{Syn}(M : X; Y)$ was positive whenever synergy exceeded redundancy and negative whenever redundancy was greater, it would always *underestimate* each as long as the other was present. Moreover, synergy and redundancy could precisely balance each other, leading to a cancellation of the two quantities (this will be demonstrated in Example 3.1, which

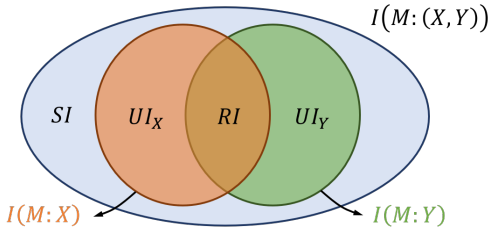


Figure 3.1: A Venn diagram representing partial information measures and their interactions, summarizing equations (3.2) and (3.3). The outer ellipse encompasses the total mutual information about M contained in X and Y, $I(M;(X,Y))$. The orange circle represents the mutual information between M and X alone, $I(M;X)$, while the green circle represents that between M and Y alone, $I(M;Y)$. The orange crescent is the information about M *uniquely* contained in X, the green crescent is that *uniquely* present in Y, the brown overlapping region is the information about M *redundantly* present in both X and Y, and the blue region outside both circles is the *synergy* between X and Y about M.

appears later). Some of these issues were recognized by Schneidman et al. [93], but there were no better ways of quantifying synergy and redundancy at that time.

Next, we present a more current understanding of synergy, arising out of the literature on Partial Information Decomposition (PID). The PID framework was first introduced by Williams and Beer [89] and subsequently advanced through a series of works, including those of Harder et al. [90], Griffith and Koch [129] and Bertschinger et al. [91] (Lizier et al. [92] provide a recent review, and Timme and Lapish [95], Pica et al. [126] show how PID may be used in the neuroscientific context).

Williams and Beer [89] suggested that synergy appears as part of a more general decomposition of the mutual information between M and (X,Y):

$$I(M;(X,Y)) = UI(M:X\setminus Y) + UI(M:Y\setminus X) + RI(M:X;Y) + SI(M:X;Y). \quad (3.2)$$

The four terms on the right hand side are respectively the information about M *uniquely* contained in X and not in Y, that uniquely contained in Y and not in X, that *redundantly* expressed in both X and Y, and that which only arises out of a *synergistic* combination of X and Y. Example 3.1, which appears shortly, shows a joint distribution with each type of information.

Intuition demands that the decomposition in (3.2) also satisfies two other constraints:

$$\begin{aligned} I(M;X) &= UI(M:X\setminus Y) + RI(M:X;Y) \\ I(M;Y) &= UI(M:Y\setminus X) + RI(M:X;Y), \end{aligned} \quad (3.3)$$

since we expect that the total information about M present in X, $I(M;X)$, is the sum of the information about M uniquely present in X and the information redundantly encoded in both X and Y (which can be extracted from either). These constraints are summarized in the Venn diagram shown in Figure 3.1. Since we have four undefined partial information quantities and three constraints in equations (3.2) and (3.3), defining any *one* of the four partial information measures suffices to determine the rest.

Williams and Beer [89] gave a formal definition for these partial information quantities, which is often called the Minimum Mutual Information (MMI) decomposition. Their

decomposition was based on a definition for the redundant information:

$$RI_{MMI}(M : X; Y) \coloneqq \min\{I(M; X), I(M; Y)\}. \tag{3.4}$$

Unfortunately, this definition has some critical shortcomings, which is seen through Example 3.1. Our work is one of the first (cf. 126) that computes more complex PID measures in neuroscientific examples, while also connecting the PID to information flow and encoding.

Example 3.1 (A simplistic example of PID)**.** Let M, X and Y be given by:

$$\begin{aligned} M &= [M_1, M_2, M_3, M_4] \\ X &= [M_1, M_3, M_4 \oplus Z] \\ Y &= [M_2, M_3, Z], \end{aligned}$$

where $M_1, M_2, M_3, M_4, Z \sim$ i.i.d. Ber$(1/2)$, and $\oplus$ refers to the exclusive-OR (XOR) of two binary variables. Thus, M has four bits of entropy, spread evenly across M_1 through M_4. Intuitively, X has 1 bit of unique information about M, encapsulated in M_1, since this is information that cannot be extracted from Y. Similarly, Y has 1 bit of unique information about M not present in X, captured by M_2. M_3 constitutes 1 bit of redundant information which can be extracted from either X or Y. Finally, M_4 is present in neither X nor Y since $M \oplus Z$ and Z are both *individually* independent of M_4. However, when X and Y are taken *together*, we can reconstruct M_4 from the combination $[M_4 \oplus Z, Z]$. Thus, X and Y have 1 bit of synergistic information about M. (Note that these values are based on intuition and not formal definitions of these quantities).

But since $I(M; X) = I(M; Y) = 2$ bits, by equation (3.4), MMI PID will find that the redundant information is 2 bits. By equation (3.3), the unique information in both X and Y is 0 bits, and from equation (3.2) the synergistic information is 2 bits. Thus, MMI PID underestimates unique information and overestimates redundant and synergistic information, relative to intuitive expectations.

Bertschinger et al. [91] proposed an improved PID definition which satisfies our intuitive expectations in a larger number of cases and has better operational foundations, coming from statistical decision theory. Their decomposition defines the *unique* information about M present in X and not in Y as

$$\begin{aligned} UI(M : X \setminus Y) &\coloneqq \min_{q \in \Delta_p} I_q(M; X \,|\, Y) \\ \text{where} \quad \Delta_p &= \{q : q(m, x) = p(m, x), q(m, y) = p(m, y)\}. \end{aligned} \tag{3.5}$$

The central intuition behind this definition arises from the following two points:

1. From equations (3.2) and (3.3), we have

$$\begin{aligned} I(M; X \,|\, Y) &= I(M; (X, Y)) - I(M; Y) && (3.6) \\ &= UI(M : X \setminus Y) + SI(M : X; Y) && (3.7) \end{aligned}$$

   Thus, the conditional mutual information is the sum of the respective unique and synergistic components.

2. Bertschinger et al. [91] argue that UI and RI should not depend on the full joint probability distribution $p(m, x, y)$, rather, they should depend only on the marginal $p(m)$ and the conditionals $p(x \mid m)$ and $p(y \mid m)$ (they justify this using a motivation from statistical decision theory). Since these marginals and conditionals are identical by definition for all $q \in \Delta_p$, UI and RI are constant over Δ_p. By taking the minimum conditional mutual information over this entire set, we are intuitively squeezing out the synergistic component (given that it is always non-negative), and defining what remains to be the unique information.

We use the definition of Bertschinger et al. [91] in all that follows.

The framework of Bertschinger et al. [91] also helps us understand where the definition for synergy used by Schneidman et al. [93] falls short. Specifically, we have

$$\mathrm{Syn}(M : X; Y) = I(M; (X, Y)) - I(M; X) - I(M; Y) = SI(M : X; Y) - RI(M : X; Y). \tag{3.8}$$

Thus, while positive values of $\mathrm{Syn}(M : X; Y)$ may indicate the presence of synergy, and negative values the presence of redundancy, neither of these implies the absence of the other. Furthermore, a zero value of $\mathrm{Syn}(M : X; Y)$ does not imply the absence of synergy and redundancy.[1] Example 3.1 also demonstrates how synergy and redundancy can precisely cancel each other in the definition of Schneidman et al. [93], leading to an overestimate of unique information and underestimates of redundant and synergistic information.

Finally, we highlight the distinction between $I(X; Y \mid M)$, which is related to conditional independence in Schneidman et al. [93], and $I(M; X \mid Y)$, which is the conditional mutual information that we use in the context of information flow.

Notation	Meaning
$H(M)$	Shannon entropy of the random variable M
$I(M; X)$	Shannon mutual information between the random variables M and X
$UI(M : X \setminus Y)$	Information about M uniquely present in X and not in Y
$RI(M : X; Y)$	Information about M redundantly present in X and in Y
$SI(M : X; Y)$	Information about M synergistically present between X and Y

Table 3.1: Information-theoretic notation used throughout the paper. All information quantities are measured in bits.

3.4 Synergy and Information Flow

Next, we look at how synergy is important for detecting information flow. Through examples, we show that unless we *account for synergy*, it is not always possible to track the paths along which information flows in neural circuits. Measures that account for synergy are those that use some form of *conditioning*, e.g., conditional mutual information, conditional correlation or partial correlation. Simpler measures based purely on Pearson correlation are unable to

[1] Schneidman et al. [93] call this "information independence", but we believe they were referring to an instance where both redundancy and synergy were exactly equal to zero, rather than cancelling each other out.

consistently track the paths along which information about a stimulus flows. We show this using simulated examples covering three different scales of neural information processing:

1. Neural circuits processing information encoded in single spikes
2. Circuits processing information encoded in spike trains
3. Information encoded at a population level, in the aggregate activity of multiple neurons

Our simulations are all based on networks of neurons; these rely on a reparameterized version of the Quadratic Integrate-and-Fire neuron model known as a "Theta" neuron model [130]. Particulars of the simulation setup may be found in the Methods section.

3.4.1 Information Flow in a Simple XOR Circuit

We begin with a simple demonstration of how synergy may arise in a neural circuit, using the canonical example of exclusive-OR (XOR) operations. We implement an XOR operation using a network of three theta neurons. This is achieved as follows:

$$M \text{ XOR } Z = (M \text{ OR } Z) \text{ AND NOT } (M \text{ AND } Z) \tag{3.9}$$

The XOR operation is realized by dividing it into one OR and two AND operations, each of which is implemented using a theta neuron with synaptic weights set appropriately, in relation to the neuron's threshold (this is depicted in Figure 3.2a). In the above, M is a "message" (which can be thought of as a stimulus) that the network is trying to encode or convey, while Z is a noise variable representing an independent signal, or internal neural variability. M and Z are both encoded in the form of single spikes, i.e., if $M = 1$ in a given trial, a neuron receiving M as input would receive a single spike, and if $M = 0$, it would receive no spike.

In all that follows, the three-unit XOR network is condensed into a single "node" for the purpose of examining information flow.[2] Using this XOR node, we design a circuit with the intent of demonstrating how accounting for synergy is essential when inferring information flow (shown in Figure 3.2b). This circuit consists of three nodes: the first node X_1 performs an XOR of M and Z; the second node X_2 acts as a delay element and preserves the noise variable Z along a separate path; and the third node X_3 performs another XOR operation, removing the noise Z and recovering the message M. We see that effectively, information about the message M is never lost in the circuit. However, at an intermediate time step, it is preserved synergistically in the form $[M \oplus Z, Z]$ along two separate edges. Thus, when inferring information flow at this time instant, it is necessary to account for synergy in the system. Otherwise, we lose track of where information about M is present in the system.

Figure 3.2c shows the response of the network for all possible values of M and Z. If we analyze the information flow of M on different transmissions of this network, we find that around $t = 5$ms, spikes corresponding to M and Z are seen, each independent of the other and equally likely to be zero or one (corresponding to $H(M) = 1$ bit). Around $t = 24$ms, we find that X_3 shows perfect correlation with M, so that $I(M; X_3) = 1$ bit. Around $t = 14$ms, however, X_1 and X_2 both show no dependence on M, i.e., $I(M; X_1) = 0$ and $I(M; X_2) = 0$.

If we are aware of the underlying anatomy, this should strike us as perplexing, since M appears to have bypassed X_1 and X_2 to arrive at X_3, even though there are no other

[2] Evidence exists, even in the neuroscientific literature, to indicate that single neurons may compute XOR's in their dendrites [131].

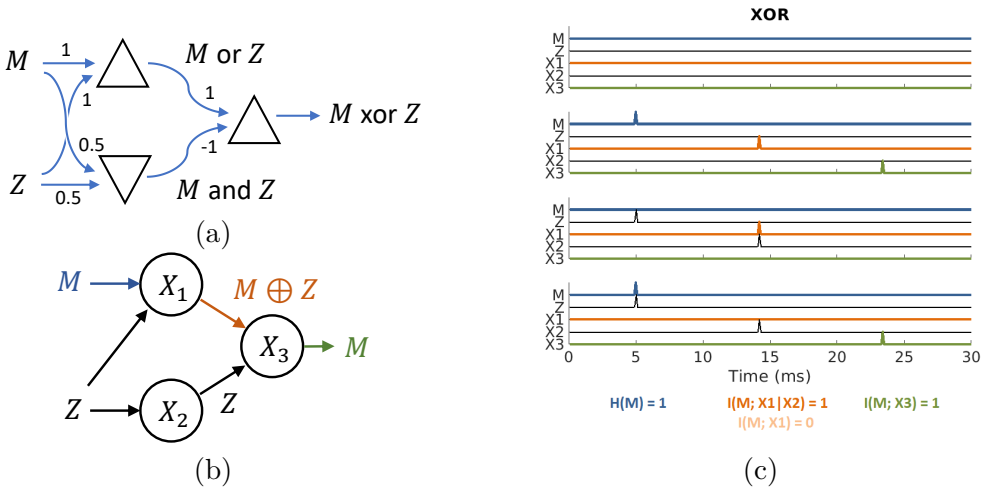
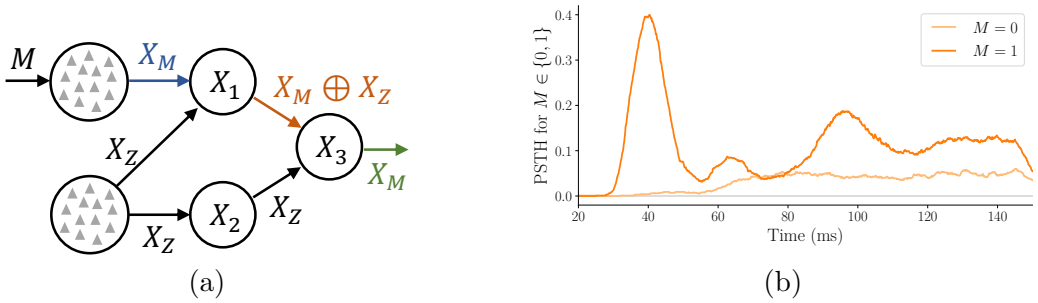


Figure 3.2: A demonstration of how synergy is essential for inferring information flow, using a simple XOR network. (a) A depiction of the circuit designed to perform an XOR operation. Here, all neurons have a threshold of 1, and the numbers on edges are an indication of approximate synaptic weight. The upper neuron on the left is excitatory and fires if either M or Z is active, thus it performs an OR operation. The lower neuron on the left is inhibitory and fires only if both M and Z are active, and therefore performs an AND operation. Since the lower neuron is inhibitory, the neuron to the right fires only if the upper neuron fires and the lower one does not; effectively resulting in an exclusive-OR of M and Z. (b) The setup of the various nodes used in this example. Here, X_1 and X_3 are XOR circuits shown in (a), while X_2 is a delay element consisting of excitatory neurons. The function of this circuit is to create a representation of M that is purely synergistic, comprised of the combination $[M \oplus Z, Z]$. (c) The behavior of the circuit in (b) for all four combinations of the binary variables M and Z. Firstly, observe that X_1, X_2 and X_3 behave as intended. Secondly, the activity of X_1 and X_2 are both individually independent of M, while the activity of X_3 is identical that of M, barring axonal delays. We observe information flow of M in X_1 around $t = 14$ms only when *conditioning* on X_2, in other words, when accounting for synergy.

nodes in the system. The issue here lies with how we evaluate the "presence of information about M", which does not account for possible synergy between X_1 and X_2. Our previous theoretical work [49] proposed to resolve this issue by conditioning on X_2 when examining the information flow of M on X_1. Our definition of M-information flow (see 49, Definition 4) states that an edge carries information flow about M if its transmission depends on M, allowing for conditioning on other concurrent transmissions. In fact, the use of this definition reveals the flow of information about M on X_1, since $I(M; X_1 \mid X_2) = 1$ bit in this example.

Interestingly, this also implies that X_2 has information flow of M, since $I(M; X_2 \mid X_1) = 1$ bit. Indeed, any time a transmission synergistically contributes to the information flow of M, our definition considers it to have information flow. The reason for this is that it is not easy to determine *which* of two edges that synergistically contribute to information about M is "actually" responsible for carrying that information. Indeed, this precise question was addressed in much greater theoretical depth in another work of ours [123]. For the purposes of our discussion here, it suffices to note that this is not necessarily undesirable, and that there are pruning-based methods that can, in most cases, remove such edges if the need arises.

The idea that synergy can be expressed using an XOR operation, which can be operationalized using neuron models, is hardly new. Such examples, in their simplest setting, have



(a) (b)

Figure 3.3: (a) A schematic of the circuit used to create synergy in the system, similar to the one in Figure 3.2b. Here, X_M and X_Z are spike trains representing M and noise respectively, while X_1, X_2 and X_3 are the same as in Figure 3.2b. (b) The PSTH of X_M for $M = 0$ and $M = 1$; note that X_M differs significantly for the two values of M (i.e., X_M represents M) only in the time intervals 30–50ms and 90–150ms.

been shown before by Timme and Lapish [95] and more broadly in the PID literature [90, 91]. However, the idea that synergy plays an important role in inferring information flow has not been pointed out before, until our earlier work [49]. What we have shown here is that the interplay of synergy and information flow may arise in real neural circuits.

We should also note that examples where accounting for synergy is essential are not limited to the case of XOR's. Such situations may also arise in simpler settings, such as with excitatory addition followed by inhibitory cancellation. There are plenty of biological examples of such self-cancellation circuits, the most common being those of efference copies [132], or corollary discharge [133]. There is also reason to believe that synergistic encoding is the product of a certain kind of information mixing, which is common in compression and error-control contexts (we will see one such example that uses grid cells in a later section). Such information mixing is likely to arise in the olfactory cortex, where there is evidence of a compressive-sensing–type circuit [134]. Lastly, coming back to XOR's, there has even been recent evidence to show that dendrites may compute XOR's [131].

3.4.2 Information Flow in a Spike Train Encoding Model

The previous example was simplistic by design, to make the argument for synergy and information flow succinctly. It was designed using single spikes in order to work cleanly with the XOR circuit model, which required somewhat precise timing of spike arrival to allow for cancellation of Z. Next, we consider a more complex scenario, to show that our conclusions about synergy and information flow are not affected by the aforementioned simplifications. We model a situation where the stimulus M is encoded by a sensory system in the form of a *spike train*. M is then passed on to a downstream region for further processing, and we are interested in how information about M flows in this downstream network.

Once again, to examine the importance of synergy, we take the downstream circuit to be the XOR network we examined before, where information about M is corrupted by noise Z, and is subsequently recovered through some form of cancellation. Therefore, the circuit being analyzed is the same as in Section 3.4.1, but M and Z are now encoded using spike trains rather than as single spikes. We use X_M to denote the spike train for M and X_Z to denote that for Z. The spike trains are generated by a randomly connected network of theta

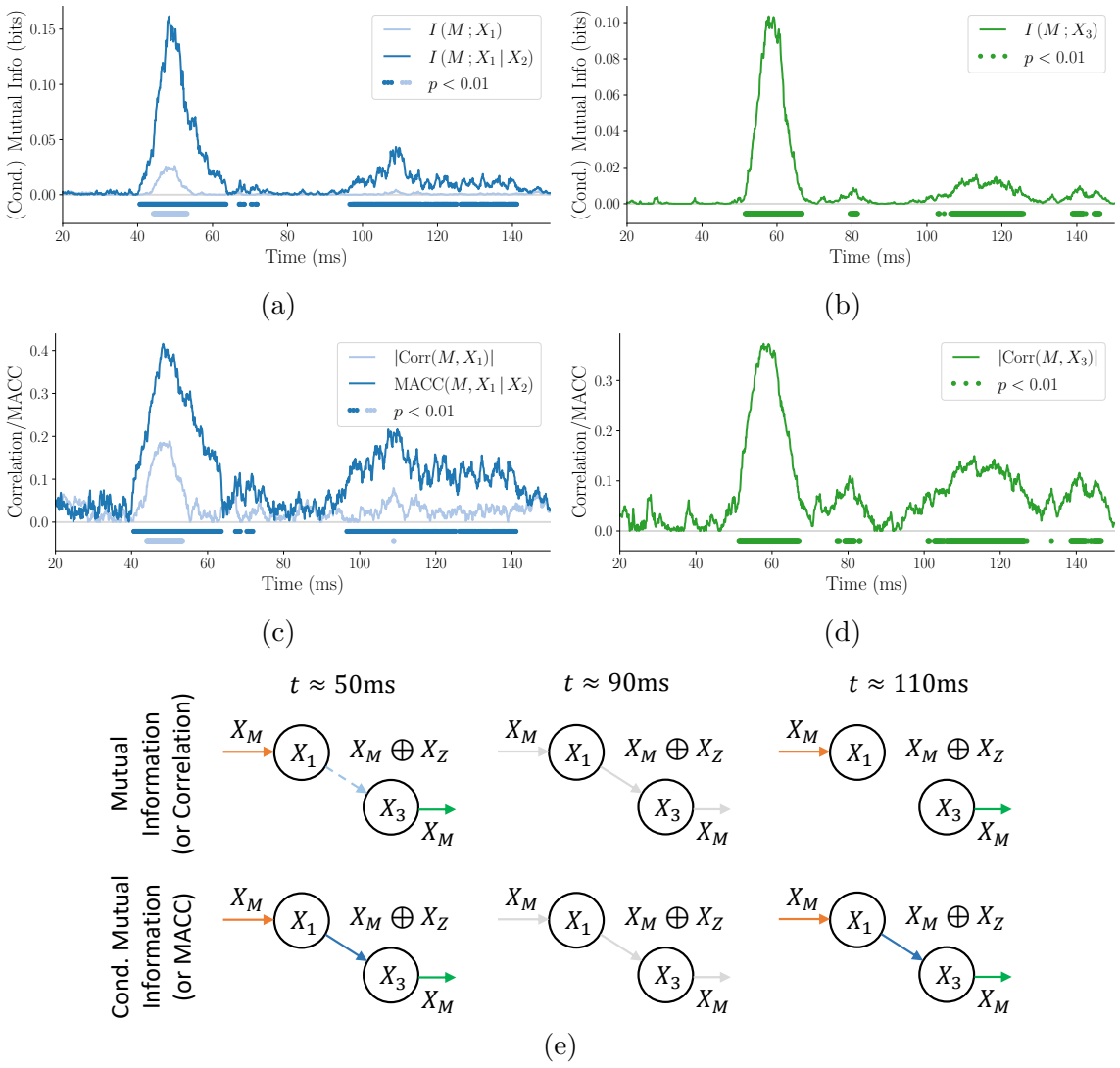


Figure 3.4: (a) Mutual information (dark blue) and conditional mutual information (light blue) between M and X_1 (conditioned on X_2). The dots beneath the graph show time instants at which the mutual information is significant ($p < 0.01$). During time instants at which M is discernible from X_M, only conditional mutual information reveals the presence of M. (b) Mutual information between M and X_3, showing recovery of information about M. (c,d) The same as in (a,b), but using correlation and MACC in place of mutual information and conditional mutual information respectively. We see that MACC is an effective substitute for conditional mutual information in this case, showing that one may be able to account for synergy without a significant burden on estimation, as long as the mode of dependence is not strictly nonlinear. (e) Estimated flows based on mutual information (or correlation) and conditional mutual information (or MACC): around the first peak at $t \approx 50$ms, MI gives significance only partially; around $t \approx 90$ms, M is not discernible, so all edges have zero flow; around the second peak at $t \approx 110$ms, only conditional mutual information and MACC reveal the flow, whereas mutual information and correlation show a break in the information path. (Summary) Around the second time interval when M is discernible (approximately 100-130ms), we see from (b) and (d) that X_3 encodes M; however, when using plain mutual information (i.e., without accounting for synergy), (a) and (c) show us that X_1 and X_2 do not. Therefore, the failure to account for synergy makes it appear as though M momentarily disappears from the network before reappearing at X_3, as seen in the first row and third column of (e).

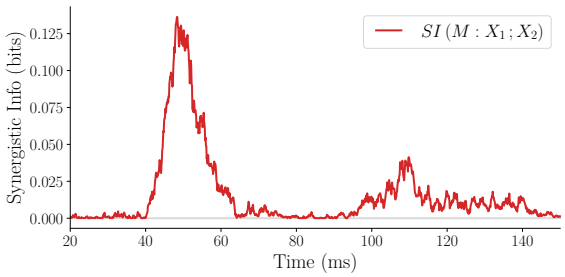


Figure 3.5: An estimate of the synergy between X_1 and X_2 about M in this system. As we might expect, the synergy is large and nearly equal to $I(M; X_1)$ at time instants when M is discernible.

neurons with balanced excitation and inhibition, as shown in Figure 3.3a. The spike train X_M is the output of a single neuron from this network which encodes the value of M at most time instants. The spike train X_Z is taken to be very noisy, having an approximately equal likelihood of firing and not firing within every 10 ms time interval.

In this case, we first note that the message M is discernible from the spike train X_M only at certain distinct intervals of time. This is seen in the peristimulus time histogram (PSTH) of X_M shown in Figure 3.3b, where M is discernible from X_M only when the light and dark curves (corresponding to $M = 0$ and $M = 1$ respectively) are separated. During these time intervals when M is discernible, we examine whether or not the relevant nodes in the XOR network reveal information flow about M.

We measure information flow about the message M in a few different ways: first, we use the measure proposed in our earlier work [49]. This is depicted in Figures 3.4a,b: observe that the transmissions of X_3 show statistically significant dependence with M in the 50–65ms and 100–125ms time periods in Figure 3.4b. This corresponds nicely with the time intervals where M is discernible in Figure 3.3b (approximately 30–50ms and 90-110ms respectively). Figure 3.4a shows that simple mutual information, $I(M; X_1)$, does not reveal statistically significant information flow about M in the transmissions of X_1, especially in the 95-115ms time interval. However, *conditioned* on X_2, we see strong conditional dependence in the transmissions of X_1, once again proving the importance of accounting for synergy when inferring information flow.

Since (conditional) mutual information is a difficult quantity to estimate in general, we also show how the same inferences can be obtained using a simpler adaptation of this measure. We use a correlation-based approximation of conditional mutual information that we call the mean absolute conditional correlation (MACC), defined as

$$\mathrm{MACC}(M : X; Y) \coloneqq \mathbb{E}_y \big| \rho(M, X \,|\, Y = y) \big|, \tag{3.10}$$

where $\rho(M, X \,|\, Y = y)$ refers to "conditional correlation", i.e., the correlation between M and X in the conditional distribution $p_{M,X\,|\,Y}(m, x \,|\, y)$, and the expectation in (3.10) is taken with respect to the marginal distribution of Y, i.e., $p_Y(y)$.

Figures 3.4c,d show analogous results to those we see for mutual information. Only upon conditioning are we able to track the paths along which information flows in this network. In particular, during the time interval 100–120ms, we see evidence of M at the output of X_3, but it is not clear how it got there when we use only mutual information to examine X_1.

Figure 3.4e shows these estimated paths of information (or lack thereof) at three different time instants.

We also show an estimate of the synergy in the system using the definition of Bertschinger et al. [91], and the estimation methods of Banerjee et al. [125]; this can be seen in Figure 3.5. This is one of the first concrete demonstrations of a sophisticated partial information measure in a neuroscientific context connected to information flow (cf. 126). As we might expect, the synergy between X_1 and X_2 about M is large at precisely those time intervals when we see information flow.

3.4.3 Information Flow in a Population Model

Lastly, we examine a scenario where the binary stimulus M is encoded by a *population* of neurons P_M in their average firing rate. This scenario is meant to emulate a setting where we use a multi-electrode array to record from a few different brain regions involved in a task. We also examine a setting where we subsample a fraction of the neurons in this region as might be the case with a multielectrode array.

Once again, to show how synergy might arise in such a system, we corrupt the message in P_M with noisy inputs arising from a second population P_Z, which encodes a continuous noise variable Z. Subsequently, if the average firing rate of P_M is large, a third population P_I, which is primarily inhibitory, suppresses P_Z after an axonal delay. This is depicted in Figure 3.6a. Our primary objective is to track which edges carry information about M at various time instants in this feedback network. In this setting, we measure information flow using a different approximation of conditional mutual information, namely, partial correlation.

An important difference between this simulation and the previous one on spike-train encoding is that the message M is being constantly provided as a current input. Therefore, although there is synergistic representation of information about M during an intermediate time period, the synergy is not directly responsible for the recovery of M at a later time instant, as was the case in the previous simulation.

Figure 3.6b is analogous to Figure 3.4a for the spike train model. It shows that, unless we are using partial correlation, we do not see statistically significant information flow about M during the 90–130ms time interval. Even upon significant subsampling of these populations, we find that the average firing rate robustly encodes the message; this can be seen in Figure 3.6c, where only 10% of all neurons are being sampled. We see that partial correlation picks up synergistic encoding even when recording from just 20 random excitatory neurons in P_M.

In this example, our computations assume that we know information about M is encoded in the average firing rate of the P_M population. In practice, one may need to determine the manifold along which information about M is encoded using dimensionality reduction approaches (e.g., see Cunningham and Yu [135]). For the case of the example provided here, canonical correlation analysis would reveal that the average activity of all neurons encodes the value of M. However, as long as information is encoded in a dense subspace of neural activity, and the subsampling mechanism is random with respect to this encoding mechanism, we would expect our results of subsampling to continue to hold.

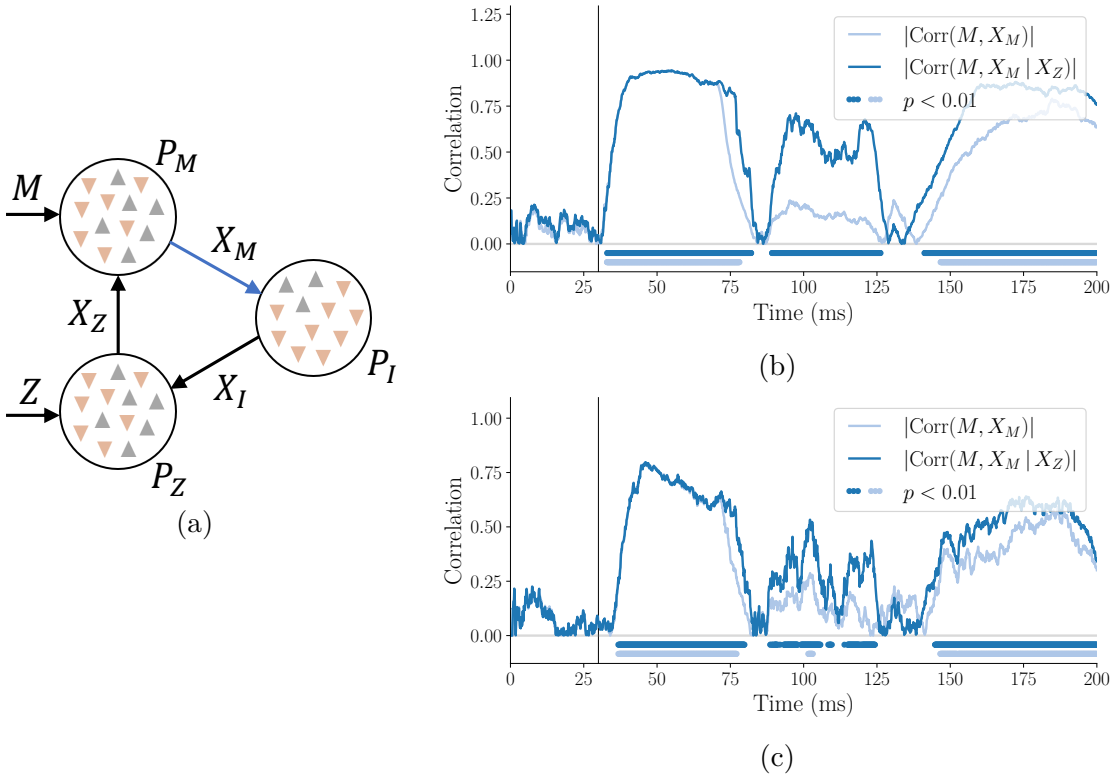


Figure 3.6: A depiction of information flow in a population encoding model. (a) A schematic representing the populations and connectivity used in our simulations. The P_M population encodes the binary message M in its average firing rate; the P_Z population encodes a continuous noise variable Z, and can greatly increase the firing rate of P_M. The P_I population is primarily inhibitory, and inhibits Z when the firing rate of P_M grows large. (b) Correlation (light) and partial correlation (dark) between M and the transmissions of P_M over time (conditioned on the transmissions of P_Z for partial correlation). Information is encoded synergistically between 90–130ms, and is statistically significant only when accounting for synergy by using partial correlation. (c) The same figure as in (b), however, only 10% of the neurons in each population were sampled while computing correlation and partial correlation. All trends we see in (b) are preserved even in (c), showing that synergy, as well as our methods of computing information flow, are robust to subsampling.

3.4.4 Remarks on our Analysis and Assumptions

A caveat to partial correlation. In the spike train model, we used MACC as an approximation for conditional mutual information to measure information flow, while in the population model, we used the more well-known partial correlation. The reason we did this is that partial correlation does not yield statistically significant information flow in the spike train model: we found that $\rho(M, X_1 \mid X_2 = 0)$ and $\rho(M, X_1 \mid X_2 = 1)$ are nearly equal in magnitude and opposite in sign; therefore, upon taking an expectation with $p(X_2)$, the two quantities cancel, leading to very small values of partial correlation, which are statistically indistinguishable from zero. This is why we use the mean *absolute* conditional correlation (MACC) instead, which takes absolute values to prevent cancellation. In general, one might want to start by trying to use partial correlation, which is a well-established method that often has easily available off-the-shelf implementations. If partial correlation reveals

information flow, then one's analysis is complete, but if not, then one cannot conclude that there is *no* flow. Instead an alternative approach based on MACC or some other approximation for conditional mutual information should be pursued.

Discontinuous information flow. Throughout our examples, we stressed on the idea that we would like to be able to track the *paths* along which information about the message M flows. In particular, we wanted to be aware of which edges and which transmissions carried information about M at every instant of time. However, there were still many time instants when it was unclear where the message was: for example, in the single spike XOR model, the time intervals between spikes revealed no information flow of the message M. In fact, information about M was still present in the network, however, it was being communicated in the form of membrane voltages along axons or dendrites and could not be seen in spikes or firing rates at the cell body. This points to a crucial hidden variable in the system: namely, the voltages on the axonal and dendritic membranes. We will only be truly able to track information flow at the resolution of axonal delays if we also measure these variables (perhaps using voltage sensitive dyes [136]). However, in practice, we find that we can get reasonably continuous and satisfactory estimates of flow (while accounting for synergy) due to random latencies and neural variability; this is evidenced in both the spike train and population encoding models.

3.5 Synergy and Encoding in Grid Cells

In this section, we present a case study on entorhinal grid cells, showing how synergy may arise in interesting (and possibly surprising) ways in biological neural systems. We begin with a short introduction on how grid cells encode information about where an animal is spatially located.

3.5.1 A Brief Introduction to Grid Cells

Grid cells are neurons in the entorhinal cortex, which are thought to encode information about where an animal is spatially located (e.g., within a room). There are a few models of how grid cells might convey such information [127, 137]; we refer to the work of Sreenivasan and Fiete [127], which is briefly described in what follows.

Each grid cell has a distinct periodic firing pattern, in that its firing rate is modulated at periodic spatial intervals. Furthermore, grid cells are organized into groups, or "modules", that all have the same periodicity (or wavelength) in their firing patterns, though their patterns may be shifted with respect to the others'. Since the cells within a module all have the same *period* but different *phase offsets* in their firing fields, these cells can be thought of as constituting a *population code* encoding the *phase* of the animal's location within that module's wavelength. As a result, the joint activity of all cells within a module can only describe the animal's position *modulo* that module's wavelength.

In order to encode location beyond a single wavelength, the entorhinal cortex consists of multiple such grid cell modules, each with their own distinct wavelength. A simple yet effective way of understanding how grid modules jointly encode information about location is to visualize the *residual uncertainty* in an animal's location, given the activity of a module.

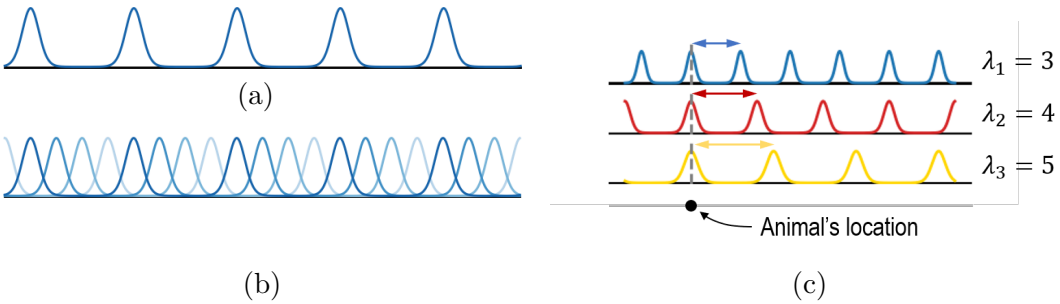


Figure 3.7: A 1D model of the activity of grid cell modules. (a) The residual uncertainty in location (also called the *conditional distribution*, or *posterior distribution* of location), given the activity of one grid module. (b) A depiction of the residual uncertainty as the animal moves along the 1D track, showing how the conditional distribution also moves along with the animal. (c) The conditional distributions of location, given the activities of three different grid modules (here, shown with wavelengths of $\lambda_1 = 3$, $\lambda_2 = 4$ and $\lambda_3 = 5$). We see that all three posterior distributions' peaks align *only* at the animal's true location. This indicates how grid modules come together to encode information about an animal's location.

A one-dimensional version of this is depicted in Figure 3.7a. If the prior distribution on the animal's location was uniform, then the posterior probability of location given the activity of a single module looks like a series of periodic peaks, separated by the wavelength of that module. Figure 3.7b shows how this posterior distribution shifts as the animal moves. The animal's location is uniquely determined only when we consider the joint activity of multiple modules: this is shown in Figure 3.7c. In particular, the posterior distributions of all grid modules *align* at the animal's true location, so that the *product* of these posterior distributions peaks at the true location.

Sreenivasan and Fiete [127] suggest that, in order to maximize the amount of space grid cells can encode, the wavelengths of different modules ought to be "co-prime" (or "incommensurate") with respect to each other. The expectation is that the total "range" that can be covered using an encoding scheme such as this is *exponential* in the number of modules, or more precisely, of the order of the *product of their wavelengths*. Sreenivasan and Fiete [127] also claim that the maximum range that may be encoded using all modules is far greater than is likely to be necessary in an animal's lifetime; thus an animal would instead encode only a restricted range, so that any additional modules are effectively used as redundancy against neural variability.

A key takeaway from this depiction is that, given the activity of a single module, there is typically still a lot of residual uncertainty which is spread across the entire possible range of movement. It is only when we put information from several modules *together* that we get a refined understanding of the animal's location. Since information about location is encoded *jointly* by multiple modules, and no one module reveals this information on its own, this system provides an excellent opportunity for understanding how partial information measures may be useful in practice.

To understand the broader applicability of the PID framework, we examine situations encompassing all three types of partial information: unique, redundant and synergistic. However, in keeping with the central theme and motivation of the paper, we will focus on how synergy arises in grid cells, and what this teaches us about both synergy and information encoding. The introduction to grid cells above, as well as Figure 3.7 may suggest that

information is primarily encoded synergistically, due to the fact that many modules come together to supply information about location: in what follows, we will see whether this is indeed the case.

3.5.2 Model setup

In this section, we briefly describe how we setup a model for a few different grid modules encoding a one-dimensional location. To keep our simulation simple and to focus on parameters of importance, we forego a spiking neuron model and instead directly model the activity of an entire grid module. We do this by assuming a conditional distribution for the residual uncertainty in location, given each module's activity. In order to account for neural variability, we let these conditional distributions have different degrees of "variance" (see methods for details).

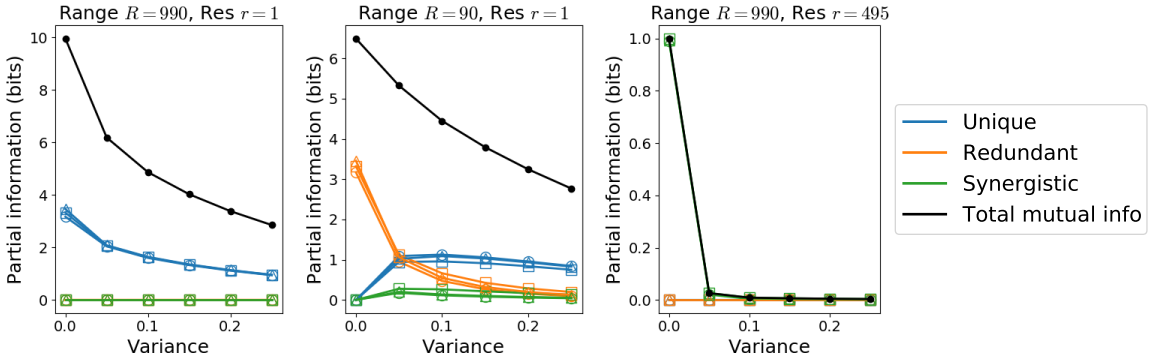
We consider a total of three modules: in our simulations and analyses, we use wavelengths of 9, 10 and 11 units; for the purpose of illustration, we use wavelengths of 3, 4 and 5 units (this is explicitly mentioned where needed). The conditional distributions are discretized to simplify the implementation of computing information measures. Once again, we use the PID of Bertschinger et al. [91], and compute partial information measures using the implementation by Banerjee et al. [125].

3.5.3 Unique, Synergistic and Redundant Information in Grid Cells

First, we examine the extent of unique, redundant and synergistic information between each module and the other two, in a setting where they encode the maximum encoding range of $9 \times 10 \times 11 = 990$. Figure 3.8a shows the unique, redundant and synergistic information in each module with respect to the other two, as a function of increasing neural variability. The main takeaway from this figure is that all of the information content is actually *unique* to each module. In other words, while each module conveys little about location by itself, there are no *synergistic* effects. Thus, the joint information from two different modules is *not* greater than their sum (it is in fact equal). Furthermore, this shows that each module reduces uncertainty about location in a way that is "orthogonal" in some sense to the others, so that there is no redundant information. This may even be expected, considering that we are operating at maximum "capacity", when encoding the maximum possible range.

Next, we examine a setting where we allow for a reduced encoding range. We consider a reduced range of $9 \times 10 = 90$, and compute unique, redundant and synergistic information about location between each module and the other two, as before (see Figure 3.8b). Here, as expected, we find that in the absence of neural variability, each module contains purely redundant information with respect to the other two (since any two other modules suffice to encode this range). On the other hand, as variability rises, the redundancy drops sharply, while uniqueness rises, and total mutual information drops more slowly. This suggests that error correction is in effect; indeed the presence of a combination of redundant and unique information could indicate some form of error correction in other settings as well. However, there is no synergy even in this setting, at any noise level.

Finally, to understand how synergy can arise in such a system, we change the "question" we are asking: instead of looking at information about *precise* location, we consider



(a) (b) (c)

Figure 3.8: Estimated partial information about location between grid modules in different settings. Each figure uses three grid modules with wavelengths $\lambda_1 = 9$, $\lambda_2 = 10$ and $\lambda_3 = 11$ for a total of 990 possible locations. (a) Quantification of unique, redundant and synergistic information about location, between each grid module and the two others, when the grid cells utilize their full encoding range of $R = 990$ units, and we are interested in resolving location at a fine scale of $r = 1$ unit. We see that all of the information is *uniquely* encoded, with each module containing only unique information with respect to the other two. This indicates that each module reduces uncertainty about location independently of, and to roughly the same as, the others. However, total information content drops sharply with increasing neural variability. (b) Estimated partial information measures when utilizing a reduced encoding range of $R = 90$ units, when we are interested in a resolution of $r = 1$ unit. Here, any two modules suffice to fully express information about location, while the remaining module provides redundancy against neural variability. We see that all of the information is redundant in the absence of noise, and that there is a mixture of redundant and unique information as neural variability increases, while synergistic information remains close to zero. Total information is lower to begin with since we are encoding fewer total locations; however, compared to the setting in (a), it also drops off less steeply with increasing variability. (c) Partial information measures for a full encoding range of $R = 990$ units, but when we are interested in a very coarse resolution of $r = 495$ units, i.e., whether the animal is in the left or right half of the room. We see that each module contains purely synergistic information about location with respect to the other two in the absence of noise. As neural variability increases, information content drops almost immediately to zero.

information about *coarse* location, for example, is the animal in the left or right half of the room? When we change the question, or in effect, the *message* under consideration, we find that synergy arises in this system. The intuition for this is explained in Figure 3.9. We find that the residual uncertainty in location, given the activity of one module, spans both left and right halves of the room equally. Thus the uncertainty in left-vs-right remains as it did before we knew the activity of a module. The same applies when we know the activities of any *pair* of modules. Indeed, it is only when the activities of all three modules are known that the residual distribution collapses into one of the two halves of the room. This is indeed close to the canonical example for synergy: each module individually gives little to no information about a message, but jointly they explain everything about the message.

The main takeaway from this analysis is that synergy can arise in a circuit in unexpected ways: in this instance, changing the message changed how it was represented between different modules.

We believe that measuring partial information quantities may help distinguish between hypotheses such as that of Sreenivasan and Fiete [127] and Wei et al. [137].

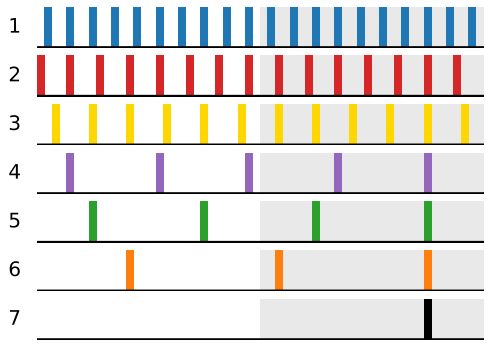


Figure 3.9: An explanation of why synergistic information behaves in the way shown in Figure 3.8c. The first three rows show a discretized representation of the residual uncertainty in location, given the activity of each module (for ease of explanation, we use $\lambda_1 = 3$, $\lambda_2 = 4$ and $\lambda_3 = 5$ respectively). The net residual uncertainty is roughly equal in both left and right halves of the room (white and grey regions), indicating that each module by itself gives no information about left-vs-right. The next three rows show the residual uncertainty, given the activities of *pairs* of modules: (3, 4), (3, 5) and (4, 5) respectively. Once again, the net residual uncertainty is approximately equal in the two halves of the room, and would only get more evenly spread as the wavelengths increase in magnitude. This indicates that even pairs of modules, taken together, do not convey much information about whether the animal is in the left or right half of the room. It is only when all three modules' activities are accounted for that the distribution collapses into one half of the room, providing the one bit of information about whether the animal is in the left or right. This is a quintessential example of synergistic encoding, where individual or pairwise modules give little to no information about left-vs-right, but together, they completely specify this information.

3.6 Methods

3.6.1 Details of Simulations for Information Flow

3.6.1.1 Neuron model

Simulations used the theta model for neurons [130]. The theta model is a change of variables from the standard Quadratic Integrate-and-Fire (QIF) model that expresses the voltage in terms of an angle on the unit circle, $V(t) = \tan(\theta/2)$. A neuron spikes when $\theta = \pi$ ($V \to \infty$) and is reset by subtracting 2π ($\theta \to -\pi$, $V \to -\infty$), thereby removing the discontinuity of the QIF model. Each neuron is governed by the set of differential equations

$$\frac{d\theta^k}{dt} = 1 - \cos(\theta^k) + (1 + \cos(\theta^k))(I_0^k + w_e^k s_e - w_i^k s_i) + \sigma\epsilon \tag{3.11}$$

$$\frac{ds_k}{dt} = \frac{1}{\tau_k}(-s_k + f_k) \tag{3.12}$$

where k indicates the type of neuron (excitatory or inhibitory), $\epsilon \sim \mathcal{N}(0, 1)$ is independently drawn for every neuron at every time step, and f_k is the number of presynaptic neurons of type k that fired at the last time step. The constant parameters are the input current I_0^k, strength of excitatory (inhibitory) synapses w_e^k (w_i^k), strength of noise σ, and synaptic decay time τ_k. The equations were numerically integrated using Euler's method ($dt = 0.1$ms).

3.6.1.2 Connectivity models

We considered three main connectivity models. As an intermediate step, three theta neurons were arranged to perform the XOR operation, as shown in Figure 3.2a. Our approach differs from the XOR gate in [95] in that we rely on different connection weights to produce the desired effect rather than a constant background inhibition. Recent results from [131] suggest that cortical dendrites possess an activation function capable of computing XOR with individual neurons, thus we consider each XOR gate as a single node regardless of its exact implementation. Two of these gates and an excitatory neuron were used to produce the first network with synergistic information. Figure 3.2b shows this network unrolled in time, where X_1 and X_3 are the XOR gates and X_2 is the excitatory neuron. Table 3.2 gives the parameter values for these neurons.

In the spike train encoding model, the binary message variable M and the noise variable Z were represented by spike trains produced by large, sparsely connected networks of theta neurons (Table 3.3). To encode $M = 1$, a constant stimulus ($I_0 \to I_0 + 0.05$) was applied to a single excitatory neuron, raising its firing rate and propagating the message through the network. On the other hand, when $M = 0$ there was no such added stimulus. In both cases, the output of a different excitatory neuron in the network was used as the spike train input for M. Figure 3.3b shows spike histograms produced for M over 1000 trials. To encode the noise variable Z, the level of noise within the network (σ) was raised so that there was an almost 50% chance of a neuron spiking within each time bin. The resulting spike trains were uncorrelated from trial to trial.

Simulations were run for 150 ms and divided into 10 ms time bins using a moving window. Correlations with the message, $\rho(M, X_i)$, were calculated for every time bin over 1000 trials, where X_i is the number of spikes produced by that node in the given time bin. To reveal synergistic information, we also calculated the conditional correlations $\rho(M, X_i \mid X_j = 0)$ and $\rho(M, X_i \mid X_j > 0)$. Since instances where $X_j > 1$ were very rare, we used these values to calculate the MACC in equation (3.10). Mutual information and conditional information were similarly estimated. Calculation of p-values was done using permutation testing ($n = 1000$), and a significance level of $p < 0.01$ is shown on all figures. Synergistic information was estimated using code provided by Banerjee et al. [125].

With population encoding, the nodes of the network became populations of theta neurons, and M and Z were encoded in the average firing rate of a population. Figure 3.6a shows the arrangement of these populations. P_M is the encoding population and receives input from M starting at 30 ms. M is again a binary message variable and determines whether neurons in P_M receive a constant input current, thereby determining the average firing rate (Table 3.4). P_Z is the noise population and receives input from Z after 70 ms. Z is independently drawn from a uniform distribution between 0 and 1, and samples are drawn until the magnitude of the correlation between M and Z across trials is less than 0.0001. Z then determines the input current to P_Z (Table 3.5). P_I is the inhibition population, and it is designed so that it easily sustains input from P_M and provides strong inhibition to P_Z, with appropriate delays to better see the flow of information (Tables 3.6 and 3.7).

Simulations were run for 200 ms with a 30 ms transient period and divided into 10 ms time bins using a moving window. As with single neuron encoding, correlations with the message, $\rho(M, X_i)$, were calculated for every time bin over 100 trials, where X_i was now the

average firing rate of the population in a given time bin. Since the firing rate is nearly a continuous variable, partial correlations (instead of conditional correlations) were calculated to reveal synergistic information. Calculation of p-values was done using the built-in Matlab function, and a significance level of $p < 0.01$ is shown on all figures.

3.6.2 Details of Simulations of Grid Cells

The conditional distribution for location given a grid module's activity was assumed to be a von Mises distribution (this is what is shown in Figure 3.7). For the purpose of computing partial information measures, however, we require discrete distributions; therefore we use a discretized version of the von Mises distribution. Neural variability affects the width of the resulting conditional distribution, and is parameterized using the circular variance of the von Mises distribution.

We compute partial information measures where the message is taken to be the discretized location (one of 990 possible locations when considering the full encoding range), and the two constituent variables are the activity of one module and the joint activity of the two others.

3.7 Discussion and Conclusion

Our simulations show that synergy may be prevalent in neural circuits: the XOR examples (both based on individual spikes as well as using spike trains) and the population coding example show that synergy is essential for inferring information flow; on the other hand the grid cell simulation shows that synergy may arise in a system when we change the message.

In other words, the grid cell example shows that even if we have previously examined and understood a system, novel stimuli may engender unexpected synergistic responses. Furthermore, unless we are able to identify and account for possible synergy (through conditioning of some form), we will be unable to track the paths along which information flows.

We also showed that synergy and its associated partial information measures of uniqueness and redundancy can be estimated in fairly complex settings using novel definitions and algorithms. The same applies to information flow: although our original definitions were based on conditional mutual information, one can often arrive at the same inferences using simpler measures such as partial and conditional correlation. Our work is one of the first (cf. 126) that computes more complex PID measures in neuroscientific examples, while also connecting the PID—specifically synergy—to information flow and encoding.

Our paper therefore makes a case for consciously examining the possibility of synergistic encoding in neural circuits and systems: because synergy may arise in ways we do not expect, because it affects our determinations of information flow, and because we now have the tools to measure it.

3.A Simulation parameters

Parameter	Detail	Value
Constant input current	to all neurons (I_0)	-0.03
Synaptic decay time	for all connections (τ)	2
Synaptic weight	Large (w_L)	0.80
	Small (w_S)	0.40
Strength of noise	XOR network (σ)	0.03
	OR network (σ)	0.04

Table 3.2: Parameter values for the single neuron encoding networks

Parameter	Detail	Value
Number of neurons	Excitatory	30
	Inhibitory	30
Probability of connection	among all neurons	0.10
Constant input current	to excitatory neurons (I_0^e)	0
	to inhibitory neurons (I_0^i)	0
Synaptic weight	from excitatory to excitatory neurons (w_e^e)	0.30
	from excitatory to inhibitory neurons (w_e^i)	0.15
	from inhibitory to excitatory neurons (w_i^e)	0.50
	from inhibitory to inhibitory neurons (w_i^i)	0.20
Synaptic decay time	for excitatory connections (τ_e)	2
	for inhibitory connections (τ_i)	8
Strength of noise	for X_M, XOR network (σ_M)	0.03
	for X_M, OR network (σ_M)	0.04
	for X_Z (σ_Z)	0.25

Table 3.3: Parameter values for the networks that generate X_M and X_Z in spike train encoding

Parameter	Detail	Value
Number of neurons	Excitatory	200
	Inhibitory	200
Probability of connection	among all neurons	0.10
Constant input current	to excitatory neurons, $M = 0$ (I_0^e)	0
	to inhibitory neurons, $M = 0$ (I_0^i)	0
	to excitatory neurons, $M = 1$ (I_0^e)	0.01
	to inhibitory neurons, $M = 1$ (I_0^i)	0.005
	Start time	30 ms
Synaptic weight	from excitatory to excitatory neurons (w_e^e)	0.30
	from excitatory to inhibitory neurons (w_e^i)	0.15
	from inhibitory to excitatory neurons (w_i^e)	0.50
	from inhibitory to inhibitory neurons (w_i^i)	0.20
Synaptic decay time	for excitatory connections (τ_e)	2
	for inhibitory connections (τ_i)	8
Strength of noise (σ)		0.10

Table 3.4: Parameter values for the encoding population (P_M)

Parameter	Detail	Value
Number of neurons	Excitatory	100
	Inhibitory	100
Probability of connection	among all neurons	0.20
Constant input current	to excitatory neurons (I_0^e)	$Z \sim \text{Uniform}(0, 1)$
	to inhibitory neurons (I_0^i)	$\frac{1}{2} I_0^e$
	Start time	70 ms
Synaptic weight	from excitatory to excitatory neurons (w_e^e)	0.30
	from excitatory to inhibitory neurons (w_e^i)	0.15
	from inhibitory to excitatory neurons (w_i^e)	0.50
	from inhibitory to inhibitory neurons (w_i^i)	0.20
Synaptic decay time	for excitatory connections (τ_e)	2
	for inhibitory connections (τ_i)	8
Strength of noise (σ)		0.10

Table 3.5: Parameter values for the noise population (P_Z)

Parameter	Detail	Value
Number of neurons	Excitatory	100
	Inhibitory	300
Probability of connection	from excitatory to excitatory neurons	0.50
	from excitatory to inhibitory neurons	0.75
	from inhibitory to excitatory neurons	0.01
	from inhibitory to inhibitory neurons	0.01
Constant input current	to excitatory neurons (I_0^e)	-0.050
	to inhibitory neurons (I_0^i)	-0.025
	start time	0 ms
Synaptic weight	from excitatory to excitatory neurons (w_e^e)	0.30
	from excitatory to inhibitory neurons (w_e^i)	0.15
	from inhibitory to excitatory neurons (w_i^e)	0.50
	from inhibitory to inhibitory neurons (w_i^i)	0.20
Synaptic decay time	for excitatory connections (τ_e)	2
	for inhibitory connections (τ_i)	8
Strength of noise (σ)		0.10

Table 3.6: Parameter values for the inhibition population (P_I)

Parameter	Detail	Value
Connection probability	P_M to P_I (excitatory to excitatory neurons)	0.10
	P_I to P_Z (inhibitory to excitatory neurons)	0.75
	P_Z to P_M (excitatory to excitatory neurons)	0.10
Delay	between P_M and P_I	10 ms
	between P_I and P_Z	15 ms
	between P_Z and P_M	0 ms

Table 3.7: Inter-population parameter values

4 M-Information Flow and Interventions in Artificial Neural Networks

You must unlearn what you have learned.

— Master Yoda

4.1 Introduction

This chapter serves to show how the M-information flow framework can be applied in the context of artificial neural networks (ANNs). It also demonstrates how we can use the knowledge about where M-information flows to intervene and reduce the dependence of the ANN's output on the message M. Thus, the empirical results in this chapter validate the usefulness of the M-information framework in practice, while also contributing to an operational interpretation of M-information flow in terms of interventions.

4.1.1 Motivation

This work is motivated by a need to develop better tools for understanding the brain, as well as artificial neural networks. In the neuroscience literature, several works have discussed what it *means* to understand the brain, which can also be extended to ANNs. For example, Marr's levels of analysis [6, 7] break down *understanding* into three[1] levels—computational, algorithmic and implementation—which, loosely speaking, are analogous to the problem statement, the algorithm and the hardware implementation of a particular task. But in ANNs, it is clear that we understand the problem statement as well as the hardware implementation; we need a more precise statement about what it means to understand the algorithm. Gao and Ganguli [3] state that "understanding will be found when we have the ability to develop simple coarse-grained models, or better yet a hierarchy of models, at varying levels of biophysical detail, all capable of predicting salient aspects of behavior at varying levels of resolution".

Two other influential works in this domain are those of Lazebnik [33] and Jonas and Kording [67]. While Lazebnik [33] focuses on developing a formal language to avoid ambiguity,

[1] Marr's original paper [6] actually had four levels of analysis, but this has since been condensed into three levels in the literature (e.g., see Churchland and Sejnowski [7]).

a takeaway for "understanding" might come from the title: we understand something if we can fix it when it is broken. Similarly, Jonas and Kording [67] argue for rigorous theory, as well as testing our methodologies on model systems with ground truth.

Distilling some of the common arguments from these papers, we propose that one way to provide an understanding of how an ANN works might comprise: (i) understanding how information from different features is *synthesized* within an ANN to produce its output; and (ii) being able to predict how *changes* to the network are likely to change its output.

Information theoretic methods have proven to be relatively successful in enhancing our intuition about how ANNs work: the information bottleneck method [57, 58], recent work on information dissipation in ANNs [138] and ideas like the InfoGAN [139] are prominent examples. Inspired by these successes, we propose a new way of using information theory to understand the internal processing of neural networks. Recently, we developed a new framework for measuring the *information flow* about a specific message within a computational system [49]. This framework has certain distinct advantages which could help address the two points for understanding ANNs mentioned above: (i) the computational system, which was designed to model the brain, can be easily extended to ANNs; and (ii) the proposed measure of information flow, called *M-information flow*, is specific to a message, and can therefore capture the flows of different features or attributes that we care about.

In this work, we extend and reinterpret the M-information flow framework for ANNs, to understand their working in the context of fair machine learning [140, 141]. We first ask whether M-information flow actually tells us something useful, i.e., can it be used to edit and change the flows in a system, towards a desired goal? To evaluate this, we measure the extent to which editing the edges that carry information flow about a protected attribute reduces bias at the output. Next, we examine whether examining the flows of two messages in concert can help us keep desirable attributes of the output while removing undesirable ones? We answer this by exploring the information flows about accuracy and bias, and examine the fairness-accuracy tradeoff for different intervention strategies, showing how these strategies work to varying extents on synthetic and real datasets.

Many previous works discuss new definitions of fairness and approaches to introducing fairness at various stages of the pipeline, as we discuss shortly. However, we emphasize here that our focus is on providing a proof-of-concept: we want to understand whether an *observational tool*, i.e. M-information flow, can predict the effects of *interventions*.[2] The fairness context is merely a specific *application domain* in which we test this hypothesis, and we don't propose to use information flow to solve the fairness problem itself. It is possible that the intervention strategies we propose can be refined to achieve the same level of performance as state-of-the-art debiasing algorithms. Indeed, it may even be advantageous to have a method that systematically tracks information flows and edits the ANN. However, this will likely require much deeper study, including a thorough comparison with existing methods, and is therefore left to future work. In this paper, we focus on *exploring a new technique*, determining how it enhances our understanding of ANNs, and how it can inform interventions for changing the behavior of an ANN in desired ways.

[2] Naturally, this will not *always* be possible, but our objective is to explore this empirically, in some common-sense simulated settings as well as on real datasets.

4.1.2 Related work

A large number of works have dealt with problems in the fields of explainability, transparency and fairness in machine learning broadly, as well as for artificial neural networks specifically. Molnar [5] provides a good summary of the different approaches taken by many of these methods for explainability. Most of these approaches seek to understand the contribution of individual features [142–144], or individual data points [145, 146] to the output. For ANNs specifically, these can also take the form of visualizations to describe what features or what abstract concepts an ANN has learned [147, 148]. There have also been a number of information theoretic approaches for measuring bias and promoting fairness in AI systems [46, 149–151].

Our approach in this paper is quite different from these prior works: we want to understand what it is about the network structure itself that leads to a certain output. We want to understand which edges carry information relevant to classification, as well as information resulting in bias, to the output. We also want to know which edges need to be changed in order to produce a desired output, e.g., fairness towards a protected group with minimal loss of accuracy.

4.1.3 Goals of this paper

We take a moment to state our goals more concretely:

1. Our primary goal, as stated in the title, is to study *whether* measuring information flows about a message can inform where we might intervene in an ANN to change how its output is affected by that message.
2. Secondly, we want to understand whether the *magnitude* of information flow lets us predict the degree to which the intervention will affect the output.
3. Lastly, we wish to examine different intervention strategies that are informed by information flows in the network, and evaluate which strategies produce the most desirable results.

Essentially, we are proposing that M-information flow framework can constitute a manner of explainability for artificial neural networks. This paper tries to validate this claim by showing that information flows give us the understanding required to intervene in an ANN.

The rest of the paper is organized as follows: in Section 4.2, we revisit the fundamentals of the M-information flow framework and show how it can be adapted for artificial neural networks. We also discuss the setup of the fairness context, which is the application domain used to evaluate the goals mentioned above. In Section 4.3, we describe how we go about empirically evaluating our goals: specifically, we discuss how we estimate M-information flow, and describe the factors involved in designing different intervention strategies. Finally, we present our results on synthetic and real datasets in Section 4.4, and conclude with a discussion of the implications of our results in Section 4.5.

4.2 Background and Problem Statement

As mentioned in the introduction, we consider the goals presented above in the context of fairness in machine learning [140, 141]. The advantage of this context is that we can show

the strengths of the information flow framework, specifically, its ability to track the flows of multiple messages: here, information flows about the protected attribute as well as about the true label.

In this section, we provide a brief introduction to our M-information flow framework [49], showing how it can be easily adapted and reinterpreted for ANNs. We also explain the setup of the fairness problem, and describe how our information flow measure is related to commonly used measures of bias against a protected attribute.

4.2.1 Adapting and Reinterpreting M-Information Flow for ANNs

First, we provide a brief introduction to the computational system model and the M-information flow definition introduced in our earlier work [49]. Then, we proceed to adapt the definition to the ANN context—in particular, making adjustments given the deterministic nature of the ANN, and providing a quantification of flow.

The M-information flow framework provides a concrete way to *define* information flow about a message M in a very general computational system. The computational system, which is modeled after the brain, is a graph consisting of nodes that compute functions and edges that transmit the results of these computations between nodes. The proposed definition of M-information flow satisfies an important property: it guarantees that we can *track* how information about M flows from the input of the system to its output.

In the original framework designed for the neuroscientific context, we think of the graph as being "feedforward in time": i.e., the computational graph is *time-unrolled* in such a way that edges send transmissions from nodes at time t to nodes at time $t+1$. Such a model is completely compatible with a feedforward neural network, where the neurons of the ANN act as the nodes, and outputs of neurons in layer t of the ANN act as the edges at time t in the computational system.

A key feature of the computational system model is how it accounts for the inherent stochasticity of the brain as well as its inputs: nodes can generate noise intrinsically, and the transmissions on the edges are considered to be random variables. In a trained artificial neural network, however, the computations at nodes is deterministic, specified completely by the weights on the edges and the neuron's activation function. We can still continue to think of the edges' transmissions as being random variables, however, since the *input data* for the neural network comes from a distribution (which could also be an *empirical* distribution, i.e., a dataset). We are now in a position to state what the M-information flow definition from our previous work [49] looks like the context of ANNs, before proceeding to adapt it for our purposes.

Definition 4.1 (Original M-information flow)**.** *Let an arbitrary edge of the neural network at layer t be denoted E_t and let the "transmission" on this edge be denoted $X(E_t)$. Similarly, let a subset of edges at layer t be denoted $\mathscr{E}'_t$ and the set of transmissions on this subset be denoted $X(\mathscr{E}'_t)$. Then we say that information about M flows on the edge E_t, i.e., edge E_t has M-information flow, if*

$$\exists\, \mathscr{E}'_t \quad s.t. \quad I(M; X(E_t) \,|\, X(\mathscr{E}'_t)) > 0. \tag{4.1}$$

Essentially, a given edge E_t can contribute information about the message M *irrespective* of other edges' transmissions (i.e., information seen in the mutual information $I(M; X(E_t))$), or *in concert* with some other subset of edges $\mathcal{E}_t'$ in the same layer t (what is known as *synergistic information*). The rationale behind the definition given above is that it counts all of these types of contributions towards the presence of information flow. Our earlier works show that such a definition is imperative, if we must have the ability to consistently track the information flow about M in all computational systems.

To adapt this definition of information flow to ANNs, we start by recognizing that the computational system from our previous work allowed each outgoing edge of a given node to carry a different transmission. In an ANN, however, the outgoing edges of a given neuron all carry the *same activation*, but with different weights; consequently, the random variables representing the transmissions are all scaled versions of each other, and have precisely the same information content. Therefore, it makes more sense to define information flows for the *activations of every node*, rather than for the transmissions of every edge. We could then use edge weights to construct a definition of information flow for edges, so as to identify the most important outgoing edges of each node and intervene in a more selective manner. Secondly, Definition 4.1 only specifies *whether or not* a given node has M-information flow. However, we require a *quantification* of M-information flow that will let us compare different nodes or edges, and decide which ones to intervene upon.

Keeping these aspects in mind, we propose M-information flow for the nodes of an ANN, followed by a quantification of M-information flow, and finally a weighted version that assigns different flows to each outgoing edge of a given node:

Definition 4.2 (M-information flow for ANNs)**.** *Let an arbitrary node of the neural network at layer t be denoted V_t, and let the activations of this node by represented by the random variable $X(V_t)$. Similarly, let an arbitrary subset of nodes at layer t be denoted $\mathcal{V}_t'$ and the set of activations of this subset be denoted $X(\mathcal{V}_t')$. Then, we say that the node V_t has M-information flow if*

$$\exists\, \mathcal{V}_t' \quad s.t. \quad I\big(M; X(V_t) \,\big|\, X(\mathcal{V}_t')\big) > 0. \tag{4.2}$$

We quantify M-information flow by taking a maximum over all subsets of nodes $\mathcal{V}_t'$ in layer t:

$$\mathcal{F}_M(V_t) \coloneqq \max_{\mathcal{V}_t'} I\big(M; X(V_t) \,\big|\, X(\mathcal{V}_t')\big). \tag{4.3}$$

Finally, if E_t is an outgoing edge of the node V_t that has weight $w(E_t)$, then we define the weighted M-information flow on that edge as

$$\mathcal{F}_M(E_t) \coloneqq w(E_t)\, \mathcal{F}_M(V_t). \tag{4.4}$$

Our definition of weighted M-information flow for the edges of the ANN is admittedly heuristic, however it has a simple rationale: an edge with a larger weight have a greater impact on the node at its receiving end.

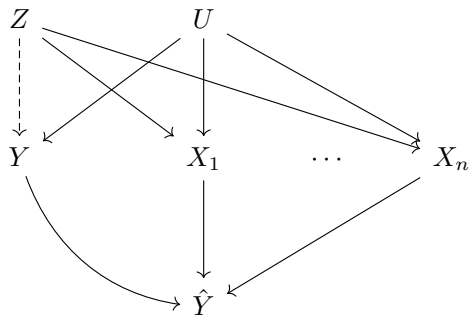


Figure 4.1: A graphical model representing the causal relationships assumed between the variables used in the fairness setup. The ANN's output $\hat{Y}$ depends on the features $\{X_i\}$ and the true labels Y, which in turn are influenced by the protected attribute Z and some latent variables U. The dashed line from Z to Y indicates that the true labels may or may not be biased.

4.2.2 Fairness Problem Setup

The central problem in the field of fair machine learning is to understand how we can train models for classification or regression without learning biases present in training data. Recent examples in the literature have shown why algorithms biased against a protected group can be of great concern [141, 152], with the rise of automated algorithms in hiring [141], criminal recidivism prediction [153], predictive policing [154], etc.

We consider the problem of training artificial neural networks for classification using datasets that have bias in their features and/or labels. The dependencies between the protected attribute (e.g., race, gender, nationality, etc.), the true labels and the features may be described using a graphical model as shown in Figure 4.1. We assume that the protected attribute Z influences the features $\{X_i\}$, and possibly the label Y, along with some other latent factors encoded in U. We then train an ANN using the labels and features to acquire the predicted label $\hat{Y}$.

Our goal is to measure two different types of flows: (i) information flow about the protected attribute, i.e., Z-information flows, which we also refer to as bias flow; and (ii) information flow about the true label, i.e., Y-information flows, which we also refer to as accuracy flow, as these are responsible for accuracy at the output. The measure of bias we consider at the output is an information theoretic version of statistical (or demographic) parity [155], which has also been used in many previous works [156]. This is because Z-information flow at the output is simply $I(Z;\hat{Y})$, since there are no other edges to condition upon.

4.3 Empirical Evaluation

In this section, we discuss how we estimate the information flow measure described earlier, and propose a few different intervention strategies for "editing" a trained neural network.

4.3.1 Estimating Information Flow

It is well known that conditional mutual information is a notoriously difficult quantity to estimate [99]. However, in our empirical study, we only consider datasets where the true labels Y and the protected attributes Z are binary, which makes estimation considerably easier. We use the following classification-based method to estimate the conditional mutual information which appears in our definition for M-information flow.

First, we attempt to construct the best possible classifier that predicts either Y or Z from the intermediate activations of each layer, say X. The generalization accuracy of this classifier indirectly tells us the extent to which information about Z (say) is present in X. More precisely, if the generalization accuracy of classifying Z from X is a, that means the probability of error in correctly guessing Z from X is $P_e \coloneqq 1 - a$. Then, from Fano's inequality [128, Ch. 2], we have:

$$H(Z \mid X) \leq H_b(P_e) + P_e \log(|\mathcal{Z}| - 1), \tag{4.5}$$

where H_b is the binary entropy function. Furthermore, since Z is binary, $\mathcal{Z} = \{0, 1\}$, hence $|\mathcal{Z}| = 2$. This simplifies the above equation to:

$$\begin{aligned} H(Z \mid X) &\leq H_b(P_e) && (4.6)\\ &= H_b(1 - a) && (4.7)\\ \Rightarrow I(Z; X) &= H(Z) - H(Z \mid X) && (4.8)\\ &\geq 1 - H(Z \mid X) && (4.9)\\ &\geq 1 - H_b(1 - a) && (4.10) \end{aligned}$$

Therefore, given any classifier which can predict Z from X with generalization accuracy a, we can compute a lower bound on the mutual information between Z and X, with a better classifier providing a tighter lower bound.

This allows us to compute all conditional mutual information quantities required by Definition 4.2 using the chain rule [128, Ch. 2]:

$$I(Z; X \mid X') = I(Z; X, X') - I(Z; X'). \tag{4.11}$$

Sometimes, we may find that the estimate for $I(Z; X, X')$ is smaller than that for $I(Z; X')$; this is because our estimates are lower bounds. Although adding variables can never decrease mutual information, in practice, adding features may reduce the accuracy of a classifier, especially if the extra feature does not contribute much useful information. In such cases, we simply truncate the conditional mutual information to zero, to prevent it from becoming negative.

4.3.2 Intervention strategies

As we stated in the title of the paper, the main goal of this work is to understand whether we can make interventions based on information flows. The purpose of these interventions is to change the neural network so as to have some desirable characteristics, such as a reduction in bias at the output, without sacrificing accuracy. We consider interventions that involve

"soft pruning", i.e., gradually reducing the weights of a neural network until the edges are completely pruned. Several factors play a role in informing how we prune edges, some of which we examine below.

Nodes or edges. Since information flow in these ANNs is primarily driven by activations at nodes, it might be the case that interventions are more effective if entire nodes are pruned or removed at a time. Alternatively, one could also make the case that pruning individual edges can have a more fine-grained impact on the information flows, and that while certain *combinations* of edges could result in increased bias, other combinations may result in improved accuracy without affecting bias at the output. Therefore, in our analyses we consider fairness-accuracy tradeoffs for both cases: pruning individual edges, as well as pruning nodes, i.e., pruning all the outgoing edges of a given node.

Pruning metric. Next, we consider what scoring metric to use when deciding which edges to prune. There are several options here: for instance, we could directly use the edges with the largest bias flows; alternatively, we could consider the edges with the largest bias-to-accuracy flow ratios. When pruning edges, we are also taking into account the weights of the respective edges when we look at the weighted flows for pruning. In this case, we cannot look at the ratio of the weighted flows, since the weights would simply cancel out; instead we must look at the *weighted ratio* of the bias and accuracy flows. We also considered the weighted ratio of *accuracy to bias* flows, to provide a point of comparison to show that the tradeoff curves are not a result of chance.

Where to stop. Lastly, we must consider *how many* node or edges to prune. This is necessarily a function of the number of features that are biased, so we cannot have an objective number in mind in advance. Therefore we consider fairness-accuracy tradeoff curves in our results that compare different numbers of nodes and edges. Another related question pertains to whether these edges with the largest bias-to-accuracy flows (say) ought to be pruned simultaneously, or whether they should be pruned sequentially, when examining where we should stop. We consider some of these various options in the results section.

On comparing edges. When pruning based on weighted information flow on edges, it is important to ensure that the weighted flows are actually *comparable*. For example, if the input features have wildly different dynamic ranges, then the weights corresponding to those features will naturally have a large variability that performs the role of scaling the feature, and will not reflect feature importance. Therefore, it is important to standardize the input features before passing them to the neural network, to have the weights correspond to feature importance rather than scaling.

4.4 Results

4.4.1 Synthetic Dataset

First, we examine information flows for a small neural network trained on a synthetic dataset (also called the `tinyscm` dataset). The synthetic dataset is designed in a manner similar to Figure 4.1. Details of the distributions and functions used are provided in Appendix 4.A. The dataset had three continuous-valued features X_1, X_2 and X_3, and a binary label Y as well as a binary protected attribute Z. Two of the three features, X_1 and X_2, were chosen to have a large bias, while X_3 and the labels Y were unbiased. Lastly, all three features

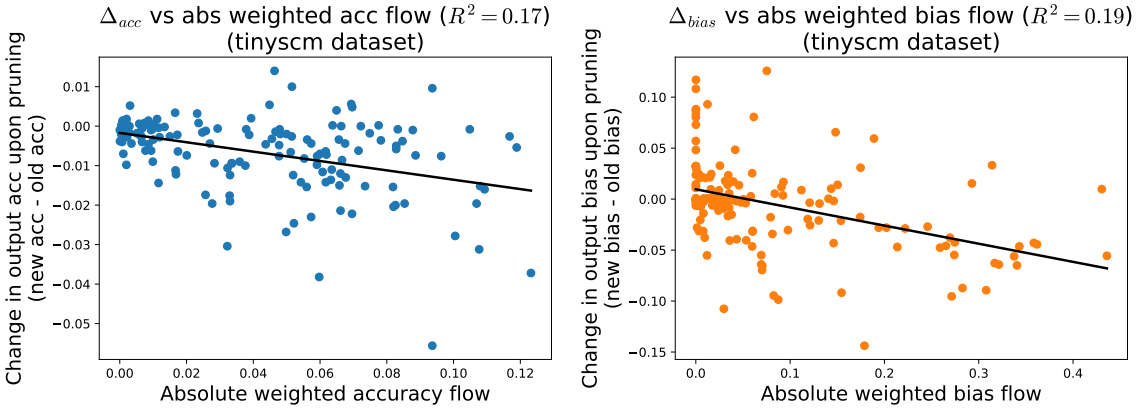


Figure 4.2: Figures showing the dependency between absolute weighted M-information flow and the change in output dependence on M for the synthetic `tinyscm` dataset. (Left) $M = Y$, so the information flows represent accuracy. (Right) $M = Z$, so the information flows represent bias. In both figures, we see that as the information flow increases, there is a larger decrease in the accuracy or bias at the output.

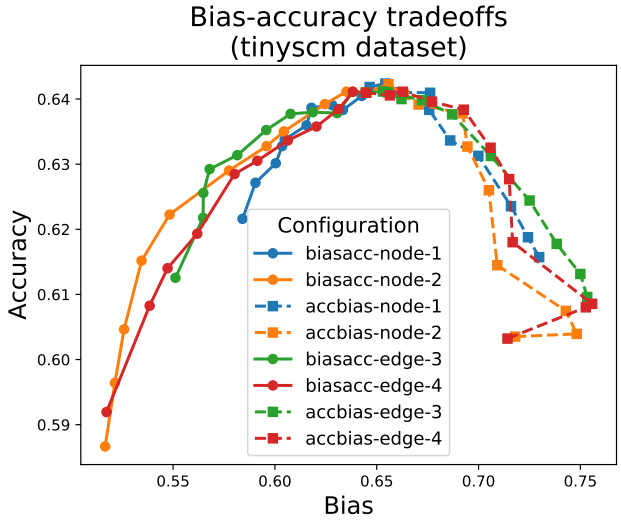


Figure 4.3: A figure showing the tradeoff between fairness and accuracy when gradually pruning nodes or edges of an ANN trained on the `tinyscm` dataset. The legend indicates the various configurations under which pruning was performed: pruning on the basis of weighted ratio of bias-to-accuracy flow (`biasacc`) or accuracy-to-bias flow (`accbias`); pruning `node`s or `edge`s; and the number of nodes or edges pruned. The general trend appears to be that pruning more lengthens the tradeoff curve without significantly shifting it in either direction. Pruning on the basis of bias to accuracy ratio causes bias to fall faster than accuracy, while pruning on the basis of accuracy to bias ratio causes bias to *increase* while accuracy falls.

independently provided information about Y, i.e., they were noisily correlated with Y, with independent noise terms.

To keep the task of estimating information flows as simple as possible, the neural network was chosen to have just one hidden layer with three neurons with ReLU activations. The output layer was a one-hot encoding of the binary $\hat{Y}$ and cross-entropy loss was used for training. The data used for training the neural network was completely separate from the

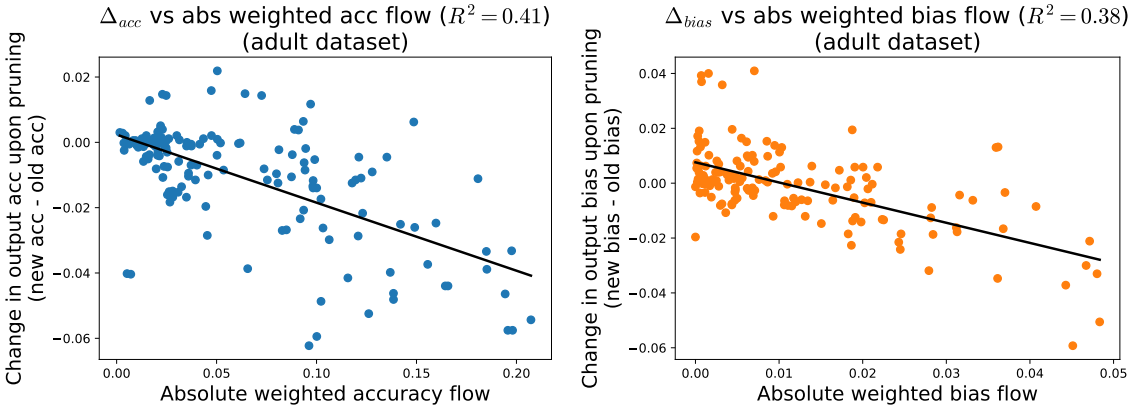


Figure 4.4: Figures showing the dependency between absolute weighted M-information flow and the change in output dependence on M for the Adult dataset. (Left) $M = Y$, so the information flows represent accuracy. (Right) $M = Z$, so the information flows represent bias. In both figures, we see that as the information flow increases, there is a larger decrease in the accuracy or bias at the output.

data used for estimating information flows on the trained network (we used a 50% split of the initial data for each component of the analysis). We used a kernel SVM to fit classifiers for the estimation of information measures (as described in Section 4.3.1). The information estimates used nested cross validation to fit the SVM hyperparameters as well as to estimate generalization accuracy. Finally, all analyses were repeated across ten neural networks trained on the same data but with different random weight initializations. Further details are in Appendix 4.B.

We first analyzed whether the extent of change in the bias or accuracy at the output upon pruning an edge was related to the magnitude of the respective information flow on that edge. To do this, we completely pruned each edge of the trained network, keeping all other edges intact, and examined the change in accuracy and bias at the output. The results of this analysis are shown in Figure 4.2. The figures clearly show that pruning edges with the largest magnitudes of weighted accuracy or bias flows tends to produce the largest change in accuracy or bias respectively at the output.

Next, we analyzed how fairness and accuracy evolve as the edges are pruned gradually, for different pruning strategies, as outlined in Section 4.3.2. The results of this analysis are presented in Figure 4.3. Once again, the results clearly show that when pruning on the basis of weighted ratio of bias to accuracy flow, the tradeoff curves are concave, indicating that bias falls faster than accuracy initially. Interestingly, pruning on the basis of weighted ratio of accuracy to bias flow shows that accuracy falls while bias *increases*. We believe this happens as a result of the reduced dependence of the output on accuracy.

4.4.2 Adult dataset

We also performed the same analyses on the Adult dataset (also known as the "Census Income" dataset) from the UCI machine learning repository [157]. The Adult dataset, which comes from the 1994 US census, consists of a mix of numerical and categorical features for classifying people with annual incomes less than and greater than $50k. For simplicity, we

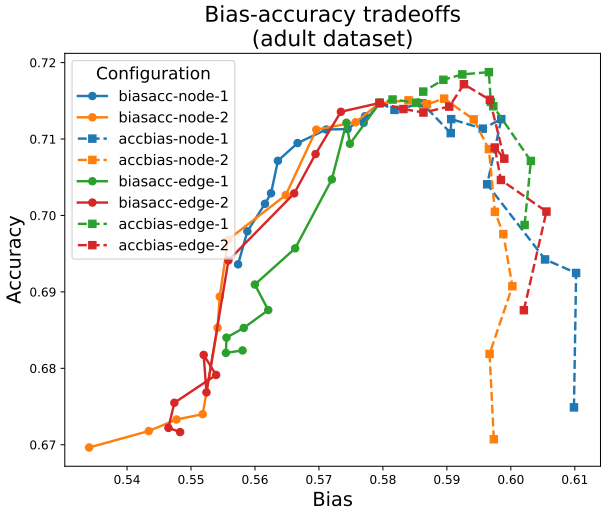
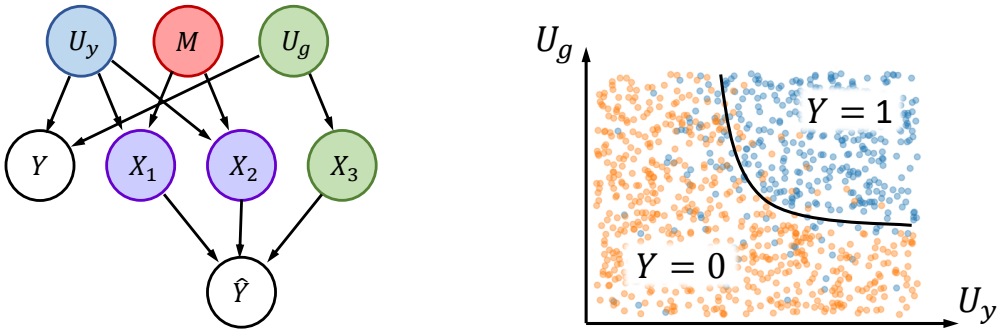


Figure 4.5: A figure showing the tradeoff between fairness and accuracy when gradually pruning nodes or edges of an ANN trained on the `tinyscm` dataset. The legend indicates the various configurations under which pruning was performed: pruning on the basis of weighted ratio of bias-to-accuracy flow (`biasacc`) or accuracy-to-bias flow (`accbias`); pruning `node`s or `edge`s; and the number of nodes or edges pruned. The general trend appears to be that pruning more lengthens the tradeoff curve without significantly shifting it in either direction. Pruning on the basis of bias to accuracy ratio causes bias to fall faster than accuracy, while pruning on the basis of accuracy to bias ratio causes bias to *increase* while accuracy falls.

use only three numerical features, viz. `education-num`, `hours-per-week` and `age`, and take `sex` to be the protected attribute. This allows us to use a small neural network, identical to the one used for the synthetic dataset. We also used only a subset of the records in order to equalize the number of individuals with high- and low-incomes and the number of male and female individuals. However, we introduced a bias in the true labels by skewing the dataset towards higher incomes among males and lower incomes among females (at a ratio of 2:1).

The results for the scaling analysis are presented in Figure 4.4, while the results for the tradeoff analysis are shown in Figure 4.5. Both results reflect the trends seen in the synthetic dataset. Interestingly, the scaling analysis from Figures 4.2 and 4.4 indicates that the ANN trained on the Adult dataset shows a stronger dependence between information flow and interventional effect than the one trained on the synthetic dataset. This may be a result of the fact that the synthetic data had two features that were *highly* biased, whereas the Adult dataset likely has much less bias in its features.

There was not a significant difference in the tradeoff curves produced by using different configurations described in Section 4.3.2, except that pruning more edges or nodes generally produced larger drops in bias and accuracy. An interesting point to note about the results on the Adult dataset is that when pruning with respect to the weighted ratio of accuracy to bias flow, the accuracy appears to *increase* slightly in some instances, along with a rise in bias, before it falls rapidly. It is unclear from the experiment why this happens, but it might suggest that the neural network was overfit to the training data, so slight changes to the weights for the test dataset actually improved its accuracy. However, it is interesting that the same interventions appear to have this effect, across ten runs of the experiment.



(a) (b)

Figure 4.6: A depiction of how the synthetic data was generated. (a) The graph corresponding to the structural equation model used to generate the synthetic dataset. (b) The relationship between the latent variables U_y and U_g, and the true labels Y, shown here for a uniform distribution over U_y and U_g for clarity. In the actual dataset, U_y and U_g were drawn from gaussian distributions.

4.5 Discussion

Our results show quite clearly that our measure of information flow can indeed inform where to intervene in a trained artificial neural network, to change its behavior at the output. Furthermore, the fairness-accuracy tradeoff curves indicate that we can use different messages to understand flows about different variables within the system, and edit the system to preserve desirable flows while removing undesirable flows. This has strong implications for neuroscience, where such an information flow measure may prove useful to understanding how information is communicated between different parts of the brain. The result also shows that we can make use of such knowledge to then intervene and change flows in ways we desire, e.g., to treat various kinds of brain diseases and disorders.

4.A Details on the Synthetic Data Model

The synthetic dataset consisted of 10,000 data points, with 5000 used to train the neural network and 5000 used to estimate information flows. The data was generated according to a structural equation model, whose graph is shown in Figure 4.6a. We start with two latent variables, U_y and $U_g \sim$ i.i.d. $\mathcal{N}(0,1)$, and the protected attribute $M \sim \text{Ber}(0.5)$, with $M \perp\!\!\!\perp \{U_y, U_g\}$. We then set Y based on a nonlinearity as shown in Figure 4.6b. The boundary of the nonlinearity is set to be

$$U_y = \frac{1}{1+U_g} - 1. \tag{4.12}$$

If $d(U_y, U_g)$ is the signed distance from any point (U_y, U_g) to this boundary, then

$$Y \,|\, U_y, U_g \sim \text{Ber}\left(\frac{1}{1+e^{-3d(U_x,U_y)}}\right). \tag{4.13}$$

X_1, X_2 and X_3 are designed to indirectly convey information about Y, by encoding U_y and U_g in a manner biased by M. X_1 and X_2 are chosen to be biased, with

$$X_1 \,|\, U_y, M \sim \mathcal{N}(0.7MU_y, 0.04), \tag{4.14}$$

$$X_2 \,|\, U_y, M \sim \mathcal{N}(0.5MU_y, 0.04), \tag{4.15}$$
$$X_3 \,|\, U_g, M \sim \mathcal{N}(0.1U_g, 0.04). \tag{4.16}$$

Note that X_1 and X_2 communicate information about U_y only when $M = 1$. Therefore, these two features are informative for classifying Y only when $M = 1$, and are completely non-informative when $M = 0$, inducing bias at the output. X_3, on the other hand, is an unbiased feature, which is equally informative for both $M = 0$ and $M = 1$.

4.B Details on Data Analysis

For both the synthetic and adult datasets, $\hat{Y}$ was estimated using a neural network with one hidden layer and three hidden neurons. The input layer consisted of the three features, which were standardized for ease of training and to have comparable weights. The output layer was a one-hot encoding of $\hat{Y}$, comprised of two neurons. The activation functions of all neurons were Leaky ReLU. Training was performed for 50 epochs with a minibatch size of 10 and a momentum of 0.9 for both datasets, and learning rates of 3×10^{-2} and 3×10^{-3} for the synthetic and Adult datasets respectively.

Information flow was estimated as described in Section 4.3.1. The classifier used was a Kernel SVM, with a Nystroem kernel approximation with 100 components, and a Stochastic Gradient Descent optimizer. We used nested cross-validation to optimize hyperparameters, while producing final information estimates on unseen generalization data points.

5 M-Information Flow and Counterfactuals

If one is to understand the great mystery, one must study all its aspects, not just the dogmatic narrow view of the Jedi.

— Chancellor Palpatine

This chapter discusses alternative definitions for information flow, starting with the additional requirement of avoiding M-information orphans (see Section 2.4.2). In so doing, it also explores the relationship between M-information flow and counterfactual causal influence, showing that the latter can also constitute an excellent way to understand information flow. However, by its very nature, counterfactual causal influence is not generally applicable in practice, except in unique circumstances where we have interventional access to all latent sources of randomness. Unfortunately, as we show, observational measures will never be able to attain the same flows as counterfactual causal influence, due to the presence of synergistic examples such as Counterexample 2.1. This chapter therefore highlights some limitations of observational measures, which we should be aware of when making interpretations about information flows.

5.1 Introduction

There is a need to understand how information flows in various kinds of computational systems: particularly in fields such as neuroscience, where we wish to understand the inner workings of the brain [8, 10–12], and in AI, where we wish to analyze, prune, or assess the trustworthiness of artificial neural networks [46, 55, 145, 158, 159]. Towards this, we recently proposed a computational model for such neural circuits, and defined a notion of information flow called M-information flow, pertaining to a specific message M in such a system [49, 160]. The primary goal of our previous work was to demonstrate that the intuitive mutual-information–based definition of flow does not satisfy very simple properties: information can "disappear" from the system and reappear at a later time instant, so we cannot always "track" how a message *flows* through the system. This necessitates a more involved definition, which uses conditioning in a particular way, to track the "information paths" along which the message flows.

However, M-information flow also has a certain counterintuitive feature: it allows for the existence of "orphans"—nodes from which M-information flows out, though none flows

in. This was partly because we chose to restrict ourselves to *observational* measures that are functions of transmissions at a *single* time instant. We did not examine counterfactual measures (which come from the field of causality [34, 35, 161] and cannot, in general, be estimated from passively observed data) and we only superficially examined how a definition based on multiple time instants can be employed.

The core contribution of the current work is an exploration of three alternative definitions of information flow. (i) A version of M-information flow with pruning, which is a function of transmissions at multiple time instants, and is a more detailed analysis of the same definition proposed in our previous work [49] (Section 5.3); (ii) A counterfactual definition that closely matches our intuition in many cases, but cannot be estimated using passively observed data (Section 5.4); (iii) A modified M-information flow definition based on conditional mutual information, where we allow for functions to be applied to transmissions prior to conditioning—as stated, this is not computable in general, but might be more appealing in some settings (Section 5.6). We also prove an impossibility result: no observational measure of information flow that guarantees information paths can match counterfactual causal influence exactly; it *will*, in some instances, award information flow to edges that counterfactual causal influence will not (Section 5.5).

We note that all three proposed definitions allow us to track information paths while also not having orphans (possibly after pruning). However, each definition we examine has its own shortcomings, giving rise to non-intuitive paths in at least some cases. Recognizing and understanding these shortcomings can help us determine which definition is better suited for a particular purpose. Despite no definition being ideal, this systematic framework lends itself much better to drawing clear interpretations than classical tools used in neuroscience, such as Granger Causality. We revisit this point in Section 5.7.

5.2 Background

We begin with a short recap of our computational system model and the definition of information flow about a message M discussed in [49]. We also restate two important properties of our M-information flow definition: firstly, that it guarantees the existence of "information paths" along which information about the message flows in the system; and secondly, that it suffers from "orphans". The definitions as well as the counterexample in this section are largely replicated from our previous work [49] with only minor modifications, in order to keep this paper self-contained.

5.2.1 The Computational System Model

Definition 5.1 (Time-unrolled graph)**.** *Let $\mathcal{G}^* = (\mathcal{V}^*, \mathcal{E}^*)$ be a fully-connected directed graph with N nodes, i.e., $\mathcal{V}^* = \{1, 2, \ldots, N\}$ and $\mathcal{E}^* = \mathcal{V}^* \times \mathcal{V}^*$. Also, let $\mathcal{T} = \{0, 1, \ldots, T\}$ be a set of time indices, where T is a positive integer representing the maximum time index. Then, a* time-unrolled graph *$\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is constructed by indexing a fully-connected directed graph $\mathcal{G}^*$ using the time indices $\mathcal{T}$ as follows: (i) The nodes $\mathcal{V}$ consist of all nodes $\mathcal{V}^*$ in $\mathcal{G}^*$, subscripted by time indices $\mathcal{T}$, i.e., $\mathcal{V} = \{A_t : A \in \mathcal{V}^*, t \in \mathcal{T}\}$; (ii) The edges $\mathcal{E}$*

connect nodes of successive *times in $\mathcal{V}$, so they can be written in terms of the edges in $\mathscr{E}^*$ as $\mathscr{E} = \{(A_t, B_{t+1}) : (A, B) \in \mathscr{E}^*, t \in \mathcal{T}\}$.* □

Remarks: (i) We denote the set of all nodes at time t by $\mathcal{V}_t$, and the set of all (outgoing) edges at time t by $\mathscr{E}_t$. So, for example, we will have $A_1 \in \mathcal{V}_1$ and $(A_1, B_2) \in \mathscr{E}_1$. (ii) The original fully-connected graph $\mathscr{G}^*$ has self-edges, so the time-unrolled graph will always have an edge (A_t, A_{t+1}) in $\mathscr{E}_t$ for every node $A_t \in \mathcal{V}_t$.

Definition 5.2 (Computational System)**.** *A* computational system $\mathscr{C} = (\mathscr{G}, X, W, f)$ *is a time-unrolled graph $\mathscr{G}$ that has* transmissions on its edges *which are constrained by* computations at its nodes*. The* input nodes *of the computational system compute a function of a* message*, M. We now elaborate upon these italicized terms:*

5.2a) Transmissions on Edges

In a time-unrolled graph $\mathscr{G}$, let $X : \mathscr{E} \to \mathcal{X}$ be a function that describes what random variable is being transmitted on a given edge, i.e., $X(E)$ is the random variable corresponding to the transmission on the edge E. Here, the range $\mathcal{X}$ is the set of all random variables in some probability space.

For convenience, we define X applied to a set of edges *as the set of random variables produced by applying X to each of those edges individually, i.e., for* any *subset $\mathscr{E}' \subseteq \mathscr{E}$,*

$$X(\mathscr{E}') = \{X(E) : E \in \mathscr{E}'\}. \tag{5.1}$$

We extend the use of this notation to other functions of nodes and edges that we define, going forward.

5.2b) Computation at a Node

Let $A_t \in \mathcal{V}_t$ be a node in the time-unrolled graph $\mathscr{G}$, at some time $t \geq 1$ (recall that $t \in \{0, 1, \ldots, T\}$). Let $\mathcal{P}(A_t)$ be the set of edges entering A_t, and $\mathbb{Q}(A_t)$ be the set of edges leaving A_t. Further, let us suppose that A_t is able to intrinsically generate the random variable $W(A_t)$ at time t, where $W(A_t) \perp\!\!\!\perp W(\mathcal{V} \setminus \{A_t\})\ \forall\ A_t \in \mathcal{V}$ and $W(\mathcal{V}_t) \perp\!\!\!\perp \{M\} \cup \{X(\mathscr{E}_{t'}) : t' \in \mathcal{T}, t' < t\}$.[1] *Then, the* computation *performed by the node A_t (for $t \geq 1$) is a deterministic function f_{A_t} that satisfies*

$$f_{A_t}(X(\mathcal{P}(A_t)), W(A_t)) = X(\mathbb{Q}(A_t)). \tag{5.2}$$

Here, $X(\mathscr{E}_{t-1})$, $W(\mathcal{V} \setminus \{A_t\})$, $W(\mathcal{V}_t)$, $X(\mathcal{P}(A_t))$ and $X(\mathbb{Q}(A_t))$ all make use of the notation described in (5.1). Note that the function at a node can thereby be time-varying*. Also, the definition above does not apply when $t = 0$; this is a special case which is discussed below.*

[1] Strictly speaking, we require that M is not an ancestor of any $W(V_t)$ in the structural causal model underlying the computational system, i.e., interventions on M will not affect $W(V_t)$, even in a counterfactual setting [35].

5.2c) The Message and the Input Nodes

The message *is a random variable M, which is of interest to the observer, and for which we shall define information flow. We assume that the message enters the computational system at (and only at) time $t = 0$. We formally define the* input nodes *of the system as those nodes of $\mathcal{G}$, at time $t = 0$, whose transmissions statistically depend on the message M: $\mathcal{V}_{ip} \coloneqq \{A_0 \in \mathcal{V}_0 : I(M; X(\mathbb{Q}(A_0))) > 0\}$, where $\mathbb{Q}(A_0)$ represents the set of edges leaving the node A_0.*

As with Definition 5.2b, we define the computation performed by an input node $A_0 \in \mathcal{V}_{ip}$ as a function f_{A_0} that satisfies $f_{A_0}(M, W(A_0)) = X(\mathbb{Q}(A_0))$, and the computation performed by a non-input node at time $t = 0$, $A_0 \in \mathcal{V}_0 \backslash \mathcal{V}_{ip}$, as a function f_{A_0} that satisfies $f_{A_0}(W(A_0)) = X(\mathbb{Q}(A_0))$, where $W(A_0) \perp\!\!\!\perp W(\mathcal{V} \backslash \{A_0\})\ \forall\ A_0 \in \mathcal{V}_0$ and $W(\mathcal{V}_0) \perp\!\!\!\perp M$. □

5.2.2 Defining Information Flow

Definition 5.3 (M-information Flow)**.** *We say that an edge $E_t \in \mathcal{E}_t$ has M-information flow if*

$$\exists\ \mathcal{E}'_t \subseteq \mathcal{E}_t \setminus \{E_t\} \quad s.t. \quad I(M; X(E_t) \,|\, X(\mathcal{E}'_t)) > 0. \tag{5.3}$$

Analogously, a collection *of edges at the same time instant, $\mathcal{R}_t \subseteq \mathcal{E}_t$, is said to have M-information flow if*

$$\exists\ \mathcal{E}'_t \subseteq \mathcal{E}_t \setminus \mathcal{R}_t \quad s.t. \quad I(M; X(\mathcal{R}_t) \,|\, X(\mathcal{E}'_t)) > 0. \tag{5.4}$$

That is, we say an edge E_t (at time t) has M-information flow if, conditioned *on the transmissions of some subset $\mathcal{E}'_t$ also at time t, $X(E_t)$ has mutual information with M (here, $\mathcal{E}'_t$ includes the empty set). The rationale behind this definition is explained after Counterexample 5.1.* □

Note: Henceforth, "information flow about M" may refer to *any* measure of information flow, but "M-information flow" refers specifically to Definition 5.3.

5.2.3 The Information Path Property

Definition 5.4 (Path)**.** *In any computational system $\mathcal{C}$, suppose $\mathcal{A}$ and $\mathcal{B}$ are two disjoint sets of nodes in $\mathcal{V}$. Then, a* path *from $\mathcal{A}$ to $\mathcal{B}$ is any ordered set of nodes $\{V^{(0)}, V^{(1)}, \ldots, V^{(L)}\}$ that satisfies (i) $V^{(0)} \in \mathcal{A}$; (ii) $V^{(L)} \in \mathcal{B}$; and (iii) $(V^{(i-1)}, V^{(i)}) \in \mathcal{E}$ for every $1 \leq i \leq L$, where L is a positive integer indicating the path's length. We refer to the set $\{(V^{(i-1)}, V^{(i)})\}_{i=1}^{L}$ as the* edges of the path. □

Definition 5.5 (M-Information Path)**.** *An M-*information path *from $\mathcal{A}$ to $\mathcal{B}$ is a* path *from $\mathcal{A}$ to $\mathcal{B}$, every edge of which carries information flow about M.* □

Property 5.1 (Existence of an Information Path)**.** *In any computational system $\mathcal{C}$, suppose that at some time $t_{op} \in \mathcal{T}$, there is an "output node" $V_{op} \in \mathcal{V}$ whose outgoing edges $\mathbb{Q}(V_{op})$*

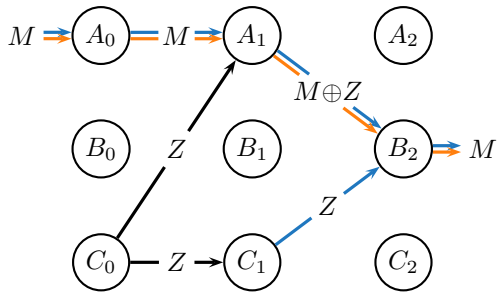


Figure 5.1: The computational system for Counterexample 5.1, which also appeared in our previous work [49] (to avoid clutter, only edges relevant to the counterexample are depicted; all other edges are still present and their transmissions are assumed to be zero). Edges in blue have M-information flow (Definition 5.3) and those in orange are M-CCI'd (as described later in Section 5.4). Observe that the edges with $M \oplus Z$ as well as Z at time $t = 1$ have M-information flow as per Definition 5.3. This results in an orphan at C_1, since the only incoming edge of C_1 does not have M-information flow.

satisfy $I(M; X(\mathbb{Q}(V_{op}))) > 0$. *Then, there must exist an* M*-information path from the input nodes* $\mathcal{V}_{ip}$ *to* V_{op}.

Theorem 5.1. *Definition 5.3 satisfies Property 5.1.*

The proof of this theorem was one of the main contributions of our earlier work, and can be found in [49]. We have reiterated the theorem statement alone for completeness.

5.2.4 The No-Orphans Property

As pointed out in our earlier work [49], Definition 5.3 also has a very non-intuitive property: the existence of orphans.

Definition 5.6 (M-information Orphan)**.** *In a computational system* $\mathscr{C}$*, a node* V_t *is said to be an* M-information orphan *if its outgoing edges* $\mathbb{Q}(V_t)$ *have information flow about* M*, but its incoming edges* $\mathbb{P}(V_t)$ *do not.* □

Property 5.2 (Absence of Orphans)**.** *M-information orphans must not exist in a computational system.*

M-information flow (Definition 5.3) does *not* satisfy Property 5.2. This is illustrated by the following counterexample.

Counterexample 5.1. Consider the computational system depicted in Figure 5.1 (note that, in order to avoid unnecessary clutter, only edges with non-zero transmissions are shown in the figure). A_0 is the input node, which has the message $M \sim \text{Ber}(1/2)$ at time $t = 0$. The system is designed to communicate M to the node B. It chooses the following strategy: at $t = 0$, A_0 transmits M to A_1. C_0 independently generates a different random number, $W(C_0) = Z \sim \text{Ber}(1/2)$, $Z \perp\!\!\!\perp M$, and sends this message to A_1, as well as C_1. A_1 then computes $M \oplus Z$ and passes the result to B_2, while C_1 sends Z to B_2. Here, the symbol "$\oplus$"

stands for XOR, the exclusive-OR operator on two bits. B_2 is thus able to recover M by once again XOR-ing its inputs, $(M \oplus Z)$ and Z.

The edges shown in blue carry M-information flow: the edges transmitting M naturally carry M-information flow; even though $M \oplus Z$ and Z do *not* statistically depend on the message, they *conditionally* depend on the message given the other (recall Definition 5.3). That is, $I(M; M \oplus Z \,|\, Z) > 0$, and complementarily, $I(M; Z \,|\, M \oplus Z) > 0$. Hence, they *also* carry M-information flow.

Observe that the node C_1 is an M-information orphan, since the edge (C_1, B_2), transmitting Z, has M-information flow, but none of C_1's incoming edges have M-information flow. □

Remark: Counterexample 5.1 essentially shows why Definition 5.3 is needed: a simpler definition that awards information flow to E_t if $I(M; X(E_t)) > 0$ would fail to identify the information path, because $M \oplus Z \perp\!\!\!\perp M$. The edge carrying $M \oplus Z$ thus plays the important role of maintaining the M-information path from A_0 to B_2 in this example.

The existence of M-information flow on (C_1, B_2) (and hence the existence of M-information orphans) might seem rather counterintuitive in a way that M-information flow on $M \oplus Z$ does not. We likely feel this way because Z was never *computed* from M. In this sense, Z lacks some kind of "functional dependence" on M, which $M \oplus Z$ does not. This point is examined in greater detail from a causality perspective in Section 5.4. In the following section, we consider a simple pruning-based mechanism and determine whether this removes orphans and edges that transmit only Z.

5.3 M-Information Flow with Pruning

One way to avoid orphans might be to consider transmissions at more than one time instant when defining information flow: for instance, we could check for information flow at a previous time instant before assigning flow to a particular edge. The principled way to do this is to traverse paths backward from the output node to the input node, while systematically pruning all "stray" paths that lead to orphans. This process is described in the form of an Information Path Algorithm in [49, Section 5]. The algorithm relies on the fact that Definition 5.3 satisfies the information path property, so that a path leading backwards from the output node to the input nodes is always guaranteed to exist.

However, while this pruning mechanism removes orphans, it does not always remove edges like Z, which do not "functionally depend" on the message M. We next present a counterexample where an edge with $M \oplus Z$ is removed, instead of the edge with Z. It should be noted that this is a highly counterintuitive example, and is very unlikely to occur as such in practice. Nevertheless it shows that even with pruning, M-information flow is not completely devoid of shortcomings.

Counterexample 5.2 (Pruning does not remove Z-edges)**.** Consider the computational system shown in Figure 5.2. Here, the message M is being communicated from A_0 to B_3 in the following manner: A_0 sends M to both A_1 and B_1, while B_0 generates $Z \sim \mathrm{Ber}(1/2)$, $Z \perp\!\!\!\perp M$, and sends it to A_1 and B_1. The node A_1 then computes $M \oplus Z$ and passes it on to

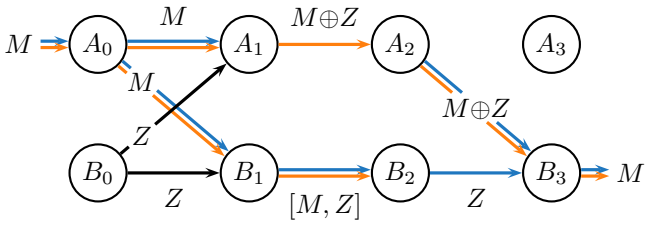


Figure 5.2: The computational system corresponding to Counterexample 5.2, which demonstrates that pruning does not always remove edges with Z. Edges in blue have M-information flow per Definition 5.3 and those in orange are M-CCI'd (as described later in Section 5.4). Counterintuitively, in this example, the edge with $M \oplus Z$ does *not* carry M-information flow per Definition 5.3.

B_3 through A_2, while B_1 simply concatenates M and Z into a vector $[M, Z]$ and sends it to B_2. B_2 then discards M, and passes on Z to B_3.

The result of this setup is that the edges shown in blue have M-information flow. In particular, the edge (A_1, A_2) carrying $M \oplus Z$, does *not* carry M-information flow: this is because $M \oplus Z$ does not depend on M by itself, and when conditioned on $[M, Z]$, naturally, M is treated as a constant and thus any mutual information with M goes to zero, i.e., $I(M; M \oplus Z \mid M, Z) = 0$. Thus, the only information path from the input node, A_0, to the output node, B_3, is the one that includes the edge (B_2, B_3), whose transmission is Z. In other words, if we were to prune edges that did not lead back to the input node A_0, we would end up removing (A_2, B_3), while (B_2, B_3), which carries Z, would remain intact. □

The existence of such a counterexample makes the information path theorem proved in [49] all the more interesting and surprising. However, it also raises several questions: on the one hand, the existence of orphans seemed counterintuitive, because their outgoing transmissions seemed to "have nothing to do with the message M"; while on the other, Counterexample 5.2 shows that even the removal of orphans does not guarantee the removal of edges with such transmissions. This makes it all the more important to focus on such edges: how are we able to *intuitively* distinguish between transmissions that in some crude sense "functionally depend" on the message M (such as $M \oplus Z$), and those that do not (e.g. Z)? We argue that the answer to this question lies in the realm of causality, in a concept known as counterfactual causal influence.

5.4 Counterfactual Causal Influence

Counterfactual causal influence [34, 35, 46, 55, 145, 158, 159, 161, 162] intuitively asks the question: for a particular *realization* of all random variables in the system, if M *alone* had been different, how would the value of some other variable have changed? This turns out to be the key to formally understanding the intuitive notion of "functional dependence" discussed above. In this section, we show that a definition of information flow based on counterfactual causal influence satisfies the information path property while at the same time having no orphans.

Definition 5.7 (M-counterfactual causal influence)**.** *The transmission on some edge E_t can be written in terms of M and all past intrinsic random variables, $\underline{W}_t \coloneqq \cup_{\tau \leq t} W(\mathcal{V}_\tau)$ as*

$$X(E_t) = g(M, \underline{W}_t), \tag{5.5}$$

for some function g. Then, $X(E_t)$ (or equivalently, E_t) is said to be counterfactually causally influenced by M (M-CCI'd) *if for some potential realization $\underline{w}_t$ of $\underline{W}_t$,*

$$\exists\, m, m' \quad s.t. \quad g(m, \underline{w}_t) \neq g(m', \underline{w}_t). \tag{5.6}$$

M-CCI constitutes a definition of information flow in that it can be treated as an indicator *of information flow about M on the edge E_t. The definition of M-CCI may also be applied in the same way to variables other than transmissions on edges.* □

Theorem 5.2. *M-CCI (Definition 5.7) satisfies Property 5.2, i.e., it does not give rise to M-information orphans. In other words, if at any node V_t, there exists an outgoing edge $E_t \in \mathbb{Q}(V_t)$ that is M-CCI'd, then there exists some incoming edge, $E'_{t-1} \in \mathcal{P}(V_t)$, which is also M-CCI'd.*

Theorem 5.3. *M-CCI (Definition 5.7) satisfies Property 5.1, i.e., it guarantees the existence of M-information paths. That is, if there is some "output node" $V_{op} \in \mathcal{V}$ that satisfies $I(M; X(\mathbb{Q}(V_{op}))) > 0$, then there exists a path from $\mathcal{V}_{ip}$ to V_{op} such that every edge of this path is M-CCI'd.*

We defer the proofs to Appendix 5.A. A brief combined proof outline for both theorems is provided below.

Proof outline for Theorems 5.2 and 5.3:

1. Link M-CCI for a single edge with that for a set of edges: if no edge in a set is individually M-CCI'd, then the set of all edges is not M-CCI'd. The converse is also true.
2. Show using Definition 5.2b that if the set of all incoming edges is not M-CCI'd, then the set of all outgoing edges is not M-CCI'd. Thus, no individual outgoing edge is M-CCI'd (by the converse in the previous point). With this, the contrapositive of Theorem 5.2 is proved.
3. Prove that if an edge is not M-CCI'd, then its transmission can have no mutual information with M.
4. Then, work backwards from the output node in Theorem 5.3 by recursively using Theorem 5.2 to show that an information path to the input nodes exists. This proves Theorem 5.3. □

As shown by the orange edges in Figs. 5.1 and 5.2, M-CCI captures the *intuitively correct* edges in these examples, e.g., $M \oplus Z$ is considered to have information flow based on M-CCI, while Z is not. This raises the question of how close we can get to M-CCI with purely observational measures, which we address in the very next section. However, we should also note here that M-CCI is not without caveats: it is not observational (i.e., cannot be estimated from passively observed data) and it can produce information paths that could be considered spurious (as we will show in Example 5.3).

5.5 The Limitations of Observational Measures

In this section, we prove an impossibility result which shows that *no* observational measure that satisfies the information path property can be made to assign information flow only to edges that are M-CCI'd. First, we formally define what we mean by observational measures.

Definition 5.8 (Observational measures of information flow)**.** *A definition of information flow is said to be* observational *if it depends only on samples of $X(\mathscr{C})$ and M. In effect, the measure depends only on the joint distribution $p(X(\mathscr{C}), M)$, which we assume can be estimated from multi-trial data.* □

In contrast, interventional and counterfactual measures require knowledge outside of the joint distribution $p(X(\mathscr{C}), M)$: we must also know how the joint distribution *changes* when one or more variables are intervened upon, or held fixed to a constant value. We next state the impossibility result, deferring its proof to Appendix 5.B.

Theorem 5.4. *Any observational definition of information flow on the edge E_t that satisfies the information path property (Property 5.1) will, in some instances, assign information flow to edges that are not M-CCI'd (Definition 5.7).*

5.6 One More Definition and an Example

Theorem 5.4 shows that observational measures are limited in that either they will not satisfy the information path property, or there will be instances where they award information flow to edges that are not M-CCI'd. However, we can ask if there are observational measures that satisfy the information path property, while at the same time providing more intuitive results upon pruning—e.g., measures that do not suffer from the counterintuitive problem discussed in Counterexample 5.2. In that spirit, we provide one more observational definition of information flow and show how it overcomes the problem discussed in Counterexample 5.2. Finally, we provide an example that brings out the differences between the three definitions presented here, and discuss their pros and cons.

Definition 5.9 (Modified M-information flow)**.** *We say that an edge E_t has modified M-information flow if there exists some subset of edges $\mathscr{E}'_t \subseteq \mathscr{E}_t \setminus \{E_t\}$, $\mathscr{E}'_t = \{E_t^{(i)}\}_{i=1}^k$ and some set of functions $\{h_i\}_{i=1}^k$ such that*

$$I(M; X(E_t) \,|\, h_1(X(E_t^{(1)})), \ldots, h_k(X(E_t^{(k)}))) > 0. \tag{5.7}$$

In other words, an edge E_t has modified M-information flow, if there exist some other edges at time t, such that when conditioned on some functions *of their individual transmissions, $X(E_t)$ has mutual information with M.* □

Every edge that has M-information flow (Definition 5.3) also has modified M-information flow, since Definition 5.9 immediately reduces to Definition 5.3 if we restrict all h_i to be identity functions. However, the opposite is not true. Consider Fig. 5.2 for example: here, all blue edges, as well as the edge (A_1, A_2), will have modified M-information flow. This is

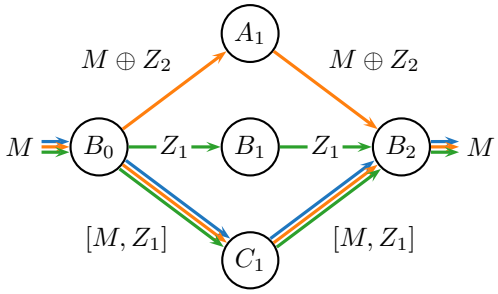


Figure 5.3: The computational system from Example 5.3, showing the differences between Definitions 5.3, 5.7 and 5.9. Edges in blue, orange and green respectively have information flow as per Definitions 5.3, 5.7 and 5.9.

because there exists a function of $[M, Z]$ (namely, $h([M, Z]) \coloneqq Z$), such that when $M \oplus Z$ is conditioned on $h([M, Z])$, we get non-zero mutual information with M. Thus, we may be avoiding some of the more non-intuitive corner cases in which M-information flow does not supply the "intuitively correct" information path.

Modified M-information flow also suffers from many of the same drawbacks as M-information flow: it still has orphans (e.g., in Fig. 5.1, only blue edges have modified M-information flow, so C_1 will be an orphan). Furthermore, as stated, Definition 5.9 is not computable, as the range of the h_i can be arbitrarily large in dimension.

Example 5.3 (All definitions are imperfect)**.** We use one last example to show that M-CCI and modified M-information flow are also not perfect, and to bring out their differences. Consider the computational system shown in Fig. 5.3. We take $M, Z_1, Z_2 \sim$ i.i.d. Ber(1/2). Note that $M \oplus Z_2$ is M-CCI'd; however, since Z_2 no longer persists in the system, all information about M has been destroyed through the XOR with Z_2. In other words, M-CCI identifies an information path which can have no computational value whatsoever.

On the other hand, the edges with Z_1 have modified M-information flow, because $[M, Z_1]$ admits the function $M \oplus Z_1$. But since Z_1 does not interact with M (save possibly *within* the node B_2), it could be argued that these edges should not carry information flow about M either.

Example 5.3 also shows that there can be M-CCI'd edges that do not have (either original [49] or modified) M-information flow; edges *not* M-CCI'd but that *have* modified (and possibly original) M-information flow; and edges that have all three. □

5.7 Discussion and Conclusion

Choosing the right definition for a particular quantity is often a hard task, and might be problem- and context-dependent, as evidenced by the multitude of definitions for entropy [83, 163]. The choice of definition is also often dictated by the trade-offs that we are willing to live with. In the case of information flow, if we are in a setting where we can examine counterfactual effects (e.g., when simulating an artificial neural network), then M-CCI provides an intuitive definition, with the caveat that it may also identify some irrelevant edges.

On the other hand, if we can only make observational measurements, then M-information flow with pruning goes a long way, save for some corner cases (such as Counterexample 5.2). We hope that these holes are also plugged when using modified M-information flow, especially in conjunction with a pruning algorithm that can remove orphans. Further work is needed to understand if there are other instances where modified M-information flow succeeds or fails in some important way.

Ultimately, it should be noted that this systematic framework for information flow, while not providing a single answer, still overcomes many of the fundamental challenges faced by classical techniques used for examining information flow. In the neuroscientific literature, Granger causality [10, 26, 27, 164] has long been used as a heuristic measure of information flow, despite several criticisms [72–75, 79–82], including the well-known fact that it is not truly representative of causation [34]. Indeed, interpreting Granger causal influence as information flow may also be questionable, as we have shown in past work [49, 82]. Given the systematic approach we have taken in defining information flow here, a natural question that arises is what connection our definition has to true causation. Our results imply that an edge has information flow about M per any of our three definitions, if *some* intervention on M can change the marginal distribution of a transmission. Similarly, edges whose transmissions statistically depend (unconditionally) on the message have information flow according to all three definitions, meaning that they are also M-CCI'd.

5.A Proofs from Section 5.4

5.A.1 Proof of Theorem 5.2

We first prove a simple lemma, which connects M-CCI for a single edge and for a set of edges.

Lemma 5.5. *For any set $\mathscr{E}'_t \subseteq \mathscr{E}_t$, if there exists some edge $E_t \in \mathscr{E}'_t$ which is M-CCI'd, then $X(\mathscr{E}'_t)$ is also M-CCI'd. The converse is also true.*

Proof. We start by enumerating the edges in $\mathscr{E}'_t$. Suppose $|\mathscr{E}'_t| =: k$. Then, we can write $\mathscr{E}'_t = \{E_t^{(i)}\}_{i=1}^k$. Now, we note that the set $X(\mathscr{E}'_t)$ is simply the collection of all transmissions in $\mathscr{E}'_t$. Therefore, we can write

$$X(\mathscr{E}'_t) = \{X(E_t^{(i)}) : E_t^{(i)} \in \mathscr{E}'_t\} \tag{5.8}$$

$$= \{g_{X(E_t^{(i)})}(M, \underline{W}_t) : E_t^{(i)} \in \mathscr{E}'_t\} \tag{5.9}$$

$$=: h(M, \underline{W}_t), \tag{5.10}$$

where $g_{X(E_t^{(i)})}$ is as defined in Definition 5.7 and h is a function that can be written in terms of the $\{g_{X(E_t^{(i)})}\}$. Now, if any one $E_t^{(j)} \in \mathscr{E}'_t$ is M-CCI'd, then there will be some set of values m, m' and $\underline{w}_t$ such that $g_{X(E_t^{(j)})}(m, \underline{w}_t) \neq g_{X(E_t^{(j)})}(m', \underline{w}_t)$. Thus, $h(m, \underline{w}_t) \neq h(m', \underline{w}_t)$, and hence $\mathscr{E}'_t$ is M-CCI'd.

Conversely, if no edge $E_t \in \mathscr{C}'_t$ is M-CCI'd, we would have

$$g_{X(E_t^{(i)})}(m, \underline{w}_t) = g_{X(E_t^{(i)})}(m', \underline{w}_t) \; \forall \; m, m', \underline{w}_t. \tag{5.11}$$

Hence, it follows that $h(m, \underline{w}_t) = h(m', \underline{w}_t) \; \forall \; m, m', \underline{w}_t$. Thus $X(\mathscr{C}'_t)$ is not M-CCI'd. This proves the lemma. □

Remark: Lemma 5.5 might seem trivial, at least in the case of M-CCI, but it is actually a crucial step in the proof of the information path property. In particular, the equivalent of Lemma 5.5 does not hold for mutual information in the converse, i.e., it is *not* true that if $X(\mathscr{C}'_t)$ has non-zero mutual information with M, then some edge $E_t \in \mathscr{C}'_t$ also has non-zero mutual information with M. Two edges' transmissions may individually have no mutual information about M, while jointly having non-zero mutual information about M. The failure of this lemma is the reason that a definition of information flow based on mutual information (as mentioned in Counterexample 5.1) does not satisfy the information path property.

Proof of Theorem 5.2. For there to be no orphans, the following must hold: at any node V_t, if there exists an outgoing edge $E_t \in \mathbb{Q}(V_t)$ that is M-CCI'd, then there exists some incoming edge, $E'_{t-1} \in \mathscr{P}(V_t)$, which is also M-CCI'd.

First, note that if all incoming edges of V_t are *not* M-CCI'd, i.e. E_{t-1} is not M-CCI'd $\forall \; E_{t-1} \in \mathscr{P}(V_t)$, then the *set* of incoming edges $\mathscr{P}(V_t)$ is not M-CCI'd. This is a direct consequence of the converse of Lemma 5.5.

Next, recall from Definition 5.2b that $X(\mathbb{Q}(V_t)) = f_{V_t}(X(\mathscr{P}(V_t)), W(V_t))$. We have already shown that $\mathscr{P}(V_t)$ is not M-CCI'd, and since M is not an ancestor of $W(V_t)$ in the structural causal model (SCM) corresponding to the computational system (see footnote 1), $W(V_t)$ is also not M-CCI'd. Thus, $\mathbb{Q}(V_t)$ is not M-CCI'd. Therefore, by Lemma 5.5, no individual outgoing edge, $E_t \in \mathbb{Q}(V_t)$, can be M-CCI'd.

Hence, by the contrapositive of the above statements, if there *is*, in fact, some outgoing edge of V_t, $E_t \in \mathbb{Q}(V_t)$, that is M-CCI'd, then there must also be an incoming edge, $E'_{t-1} \in \mathscr{P}(V_t)$, that is M-CCI'd. □

5.A.2 Proof of Theorem 5.3

Again, we first prove a simple lemma which links M-CCI with mutual information.

Lemma 5.6. *If some variable $Y := h(M, \underline{W})$ is not M-CCI'd (where $\underline{W}$ does not have M as an ancestor in the SCM corresponding to the computational system), then $I(M; Y) = 0$.*

Proof. Since Y is not M-CCI'd, we have that

$$h(m, \underline{w}) = h(m', \underline{w}) \quad \forall \; m, m', \underline{w}, \tag{5.12}$$

where $\underline{w}$ takes values in the set of possible realizations of the random variable $\underline{W}$. Thus, h is effectively independent of M, and we can write

$$Y = h(M, \underline{W}) =: h_0(\underline{W}). \tag{5.13}$$

Assuming all distributions are discrete, we can use summations to write:

$$p_{Y,M}(y,m) = \sum_{\underline{w}} p_{Y,M,\underline{W}}(y,m,\underline{w}) \tag{5.14}$$

$$= \sum_{\underline{w}} p_{Y|M,\underline{W}}(y \mid m,\underline{w}) p_{M,\underline{W}}(m,\underline{w}) \tag{5.15}$$

$$= \sum_{\underline{w}} \delta(y, h(m,\underline{w})) p_{M,\underline{W}}(m,\underline{w}) \tag{5.16}$$

$$= \sum_{\underline{w}} \delta(y, h_0(\underline{w})) p_M(m) p_{\underline{W}}(\underline{w}) \tag{5.17}$$

$$= p_M(m) \sum_{\underline{w}} \delta(y, h_0(\underline{w})) p_{\underline{W}}(\underline{w}) \tag{5.18}$$

$$=: p_M(m) c(y) \tag{5.19}$$

where in the above, δ is the Kronecker Delta function, which takes a value of 1 when its arguments are equal, and zero otherwise. In (5.16) we have made use of the fact that Y is a deterministic function of M and $\underline{W}$ to write $p_{Y|M,\underline{W}}$ as a δ-function, and in (5.17), we relied on the fact that $M \perp\!\!\!\perp \underline{W}$. Thus, we have shown that $p_{Y,M}$ can be factorized into functions purely in y and m. This implies that $Y \perp\!\!\!\perp M$, and hence $I(M;Y) = 0$. □

Proof of Theorem 5.3. Recall the theorem statement: if there is some "output node" $V_{\text{op}} \in \mathcal{V}$ that satisfies $I(M; X(\mathbb{Q}(V_{\text{op}}))) > 0$, then there exists a path from $\mathcal{V}_{\text{ip}}$ to V_{op} such that every edge of this path is M-CCI'd.

So, let us start by assuming that there is some V_{op} such that $I(M; X(\mathbb{Q}(V_{\text{op}}))) > 0$. Then, by the contrapositive of Lemma 5.6, we must have that $\mathbb{Q}(V_{\text{op}})$ is M-CCI'd. We can then repeatedly use Theorem 5.2 to find edges leading backwards in time to the input nodes. Applying Theorem 5.2 at time $t = t_{\text{op}}$, we find there must be some edge $E_{t-1} \in \mathcal{P}(V_{\text{op}})$ which is M-CCI'd. Following this edge backwards, suppose it originated from some node $V_{t-1} \in \mathcal{V}_{t-1}$. Once again, we can apply Theorem 5.2 at V_{t-1} to find another edge at time $t-2$ which is M-CCI'd. In this manner, we can find a path leading all the way back to time $t = 0$, to some node V_0. Finally, we must argue that $V_0 \in \mathcal{V}_{\text{ip}}$ based on the fact that one of its outgoing edges, say E_0, is M-CCI'd.

At time $t = 0$, Definition 5.2c implies that the outgoing edges of each node in $\mathcal{V}_{\text{ip}}$ have mutual information with M, i.e., $X(\mathbb{Q}(U_0))$ depends on M for every $U_0 \in \mathcal{V}_{\text{ip}}$. By the contrapositive of Lemma 5.6, this implies that for every $U_0 \in \mathcal{V}_{\text{ip}}$, $\mathbb{Q}(U_0)$ is M-CCI'd. Then, by Lemma 5.5, we know that there must exist some particular edge in each $\mathbb{Q}(U_0)$ which is also M-CCI'd. So we have shown that each node in $\mathcal{V}_{\text{ip}}$ has at least one outgoing edge which is M-CCI'd. But we also need to show that these are the *only* edges that are M-CCI'd, and that we cannot trace an information path all the way back to some $V_0' \in \mathcal{V}_0 \setminus \mathcal{V}_{\text{ip}}$. To show this, we once again make use of Definition 5.2c, which states that for each $U_0' \in \mathcal{V}_0 \setminus \mathcal{V}_{\text{ip}}$, $X(\mathbb{Q}(U_0')) = f_{U_0'}(W(U_0))$. Thus, each $X(\mathbb{Q}(U_0'))$ is a deterministic function of $W(U_0')$, which in turn is not M-CCI'd. Thus, $\mathbb{Q}(U_0')$ cannot be M-CCI'd, and hence no individual edge

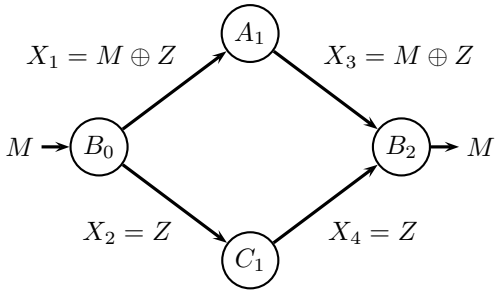


Figure 5.4: The computational system used in the proof of Theorem 5.4. Only the edges on the upper path with $M \oplus Z$ are M-CCI'd, however, the joint distribution is symmetric with respect to Z and $M \oplus Z$. As a result, any observational measure that gives information flow to $M \oplus Z$ must also give information flow to Z.

$E'_0 \in \mathbb{Q}(U'_0)$ can be M-CCI'd for any $U'_0 \in \mathcal{V}_0 \setminus \mathcal{V}_{\text{ip}}$. This proves that the information path we have traced backwards from V_{op} must lead to $\mathcal{V}_{\text{ip}}$.

Thus, there exists a path from $\mathcal{V}_{\text{ip}}$ to V_{op}, such that every edge of this path is M-CCI'd. □

5.B Proofs from Section 5.5

Proof of Theorem 5.4. Consider the computational system given in Fig. 5.4. Similar to the computational system in Counterexample 5.1, the node B_0 is trying to communicate M to B_2. However, this time, it generates Z itself, and sends $X_1 = M \oplus Z$ to A_1, while sending $X_2 = Z$ to C_1. A_1 and C_1 act merely as relay nodes, passing on $M \oplus Z$ and Z (which we label as X_3 and X_4 respectively) to B_2. Finally, B_2 computes M by XOR-ing its inputs.

The theorem statement asks us to consider any observational measure of information flow which satisfies the information path property. In the context of Fig. 5.4, the only possible information paths are (B_0, A_1, B_2) and (B_0, C_1, B_2). Therefore, any measure that satisfies the information path property will award information flow to at least one of the pairs (X_1, X_3) or (X_2, X_4).

Any observational definition of information flow would have to be a function of X_1, X_2, X_3, X_4 and M only (refer Definition 5.8). For convenience, denote $\underline{X} := [X_1, X_2, X_3, X_4] = [M \oplus Z,\ Z,\ M \oplus Z,\ Z]$. Consider the joint distribution $p_{M,\underline{X}}(m, \underline{x})$:

$$p_{M,\underline{X}}(m, \underline{x}) = \sum_{z \in \{0,1\}} p(m, \underline{x}, z) \tag{5.20}$$

$$\overset{(a)}{=} \sum_{z \in \{0,1\}} p_M(m) p_Z(z) p_{\underline{X}|M,Z}(\underline{x} \,|\, m, z) \tag{5.21}$$

$$\overset{(b)}{=} \frac{1}{4} \sum_{z \in \{0,1\}} p_{\underline{X}|M,Z}(\underline{x} \,|\, m, z) \tag{5.22}$$

$$\overset{(c)}{=} \frac{1}{4} \sum_{z \in \{0,1\}} \delta(x_1, m \oplus z) \delta(x_2, z) \delta(x_3, m \oplus z) \delta(x_4, z), \tag{5.23}$$

$$= \frac{1}{4}\Big[\delta(x_1, m \oplus 0)\delta(x_2, 0)\delta(x_3, m \oplus 0)\delta(x_4, 0) + \delta(x_1, m \oplus 1)\delta(x_2, 1)\delta(x_3, m \oplus 1)\delta(x_4, 1)\Big], \tag{5.24}$$

where in (a), we made use of the fact that $M \perp\!\!\!\perp Z$; in (b), we relied on the fact that M and Z are both Ber(1/2) random variables; and in (c), δ represents the Kronecker Delta function, and we have used the fact that $\underline{X}$ is a deterministic function of M and Z. Note that when $m = 0$,

$$p_{M,\underline{X}}(0, \underline{x}) = \frac{1}{4}\Big[\delta(x_1, 0)\delta(x_2, 0)\delta(x_3, 0)\delta(x_4, 0) + \delta(x_1, 1)\delta(x_2, 1)\delta(x_3, 1)\delta(x_4, 1)\Big], \tag{5.25}$$

and when $m = 1$,

$$p_{M,\underline{X}}(1, \underline{x}) = \frac{1}{4}\Big[\delta(x_1, 1)\delta(x_2, 0)\delta(x_3, 1)\delta(x_4, 0) + \delta(x_1, 0)\delta(x_2, 1)\delta(x_3, 0)\delta(x_4, 1)\Big]. \tag{5.26}$$

In both cases, observe that $p_{M,\underline{X}}$ is symmetric in $\underline{X}$ in a very specific way: the ordered pair (x_1, x_3) may be swapped with the pair (x_2, x_4) to no effect (i.e., $M \oplus Z$ and Z are statistically symmetric with respect to M). In the limit of large samples, any observational measure will be some functional of $p_{M,\underline{X}}$. Thus, if X_1 and X_3 are awarded information flow, so too must X_2 and X_4, by basic symmetry. This means that if the information path property holds, then all edges in Fig. 5.4 will have information flow about M according to any observational definition. Thus, Fig. 5.4 describes an instance where any observational measure that satisfies the information path property awards information flow to edges that are not M-CCI'd. □

6 M-Information Flow and Granger Causality

There is no conflict.

— Darth Vader

6.1 Introduction

This chapter discusses the relationship between Granger Causality and our notion of information flow. First, we consider whether Granger Causal influence can be used to interpret the correct *direction* of information flow in a feedback network. In this part of the chapter, when we refer to "information flow", we mean it in the intuitive sense: we introduce a specific communication-theoretic example that has a well-defined transmitter and receiver, so that the true direction of information flow is the direction in which the message is being transmitted. Using this example, we show that the direction of greater Granger Causal influence can in fact be opposite to the direction of (intuitive) information flow [82]. We also consider a more statistically rigorous analysis of this counterexample, and show that in fact, it is possible to have a Granger Causal influence that is statistically insignificant in the direction of information flow, while having a statistically significant Granger Causal influence in the opposite direction!

In the second part of the chapter, we revisit the notion of information flow we introduced in Chapter 2. We show how the communication-theoretic counterexample can be reinterpreted in the form of a computational system, and explain the information flows in the system. Our results will show that our framework for information flow finds not only the intuitively correct directions of information flow, but that the variation in the magnitude of flow over time provides insight into the communication scheme.

Finally, we discuss under what assumptions Granger Causality can actually be interpreted as capturing some form of information flow, based on our framework. We also review some prominent works in the Granger Causality and Directed Information literature to examine the outlook for such tools.

6.1.1 Motivation

This work is in large part motivated by a recent surge of interest in understanding neural circuits – the connectivity and dynamic activity of different regions of the brain – and how

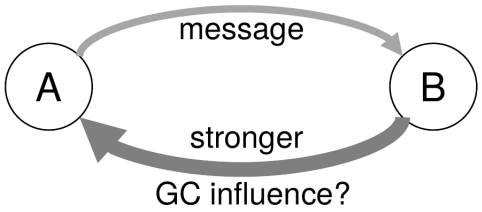


Figure 6.1: Is the direction of stronger Granger-causal influence necessarily the same as the direction in which the message is flowing?

they give rise to behavior and experience. This is evidenced by the launching of the BRAIN initiative in the US and the Human Brain Project in Europe. To quote from *BRAIN 2025: A Scientific Vision*[1], we wish to "map connected neurons in local circuits and distributed brain systems, enabling an understanding of the relationship between neuronal structure and function", clearly indicating the move towards understanding (a) the connectivity and (b) the computational function of different brain regions. While the question of how the brain computes has been of immense interest for several decades, only recently have measurement techniques become sophisticated enough to be able to simultaneously record the activity of multiple neurons, or multiple neural populations.

In order to understand how the brain performs computations, it could be useful to first understand the directions of *information flow* in various parts of the brain (e.g. [165–169] etc.). In an effort to make headway on the goals of *BRAIN 2025*, several works use Granger causality (and less often, its information-theoretic generalization – Directed Information) to understand how this information flows (e.g. [170–172]), or to acquire directed maps of functional connectivity (e.g. [69, 171, 172]). For instance, in [170], Granger causal influences that are measured between somatosensory and motor sites are said to "support the idea that somatosensory feedback provides information to the sensorimotor system that is used to control motor output". This raises the question: do these directed connectivity maps, as determined by directional causal influence measures such as Granger causality, correctly identify the directions along which information flows in the brain (see Figure 6.1)?

6.1.2 How Granger Causality is used in Neuroscience today

Several works have outlined the procedures involved in using Granger Causality to estimate causal influences in the brain ([170, 173–178]). Here, we briefly describe how Granger Causal influence is quantified, and how it is computed in these works.

Granger causality, as originally described by Granger [179], measures the level of causal influence that one process $\{X\}$ has on another process $\{Y\}$. The analysis compares the *error in predicting* the $\{Y\}$ process based on (i) simply the past of $\{Y\}$, and (ii) based on the past of both $\{X\}$ and $\{Y\}$[2]. The Granger causality metric is the ratio of these errors, encapsulating the *innovations* that the process $\{X\}$ *causally* supplies to the process $\{Y\}$. Many variants of Granger causality have also been developed, including a generalization – Directed Information (see [32, 180, 181]) – an information-theoretic quantity denoted by

[1] The BRAIN Working Group's report to the Advisory Committee to the Director of the NIH

[2] A mathematical exposition of this process appears in Section 6.3.1

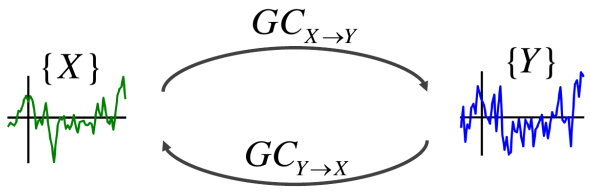


Figure 6.2: In order to determine the direction of greater causal influence, the Granger Causality metrics in the forward and reverse directions are often compared [175].

$I(\mathbf{X}^m \to \mathbf{Y}^m)$. These variants form possible alternatives for estimating the direction of causal influence, but Directed Information is a generalization of many of these metrics [181].

In order to determine the direction of *greater* causal influence, the Granger Causality metric (ratio of residual variances) from $\{X\}$ to $\{Y\}$ is often *compared* to that from $\{Y\}$ to $\{X\}$. The direction of causal influence is then taken to be the direction with the greater Granger Causality metric (e.g. [32, 170, 175], see [175] for an understanding of what physical constraints motivate this comparison). Further, this direction of causal influence is interpreted to be the direction of information flow, which is the interpretation we question in this paper. We note here that Granger's original analysis does not compare this metric on forward and reverse links, and even the stronger notion of true causality (see remark 1 in section 6.1.5) does not involve this comparison. However, this is commonly done in practice even in areas beyond neuroscience (e.g. [32, 170, 175, 182]).

We also note that many of these works use a spectral version of Granger Causality, that supplies this metric as a function of frequency. It then becomes possible to also determine the brain wave frequency at which these influences occur. However, we restrict our analysis to the simpler non-spectral version of Granger Causality, since it is sufficient for the purpose of our arguments.

While we accept that it might be possible to accurately estimate Granger Causal influence (provided measurements are taken suitably; see Section 6.1.3) and that it could be useful in many situations (see Section 6.7), interpreting the direction of greater Granger Causal influence as the direction of information flow can be erroneous, as we demonstrate in this paper.

6.1.3 A short survey of previous criticisms of Granger Causality

Several objections to the use of Granger Causality have been raised in the past. We give, here, a short overview of these and describe why our objection is novel, and possibly more fundamental in nature, at least in the context of its usage in neuroscience.

1. First, Granger Causality suffers from what we call the "hidden node problem". If two observed nodes receive causal influences from a third, latent node, then a causal influence may be detected between the observed nodes, even if they are independent of each other [183]. All nodes need to be observed, therefore, to avoid finding spurious influences.

2. Second, if the measurements from each node are differentially affected by noise, then the predicted direction of causal influence might be opposite to the true direction ([79, 80]). Measurements need to be relatively noiseless and precise in order to obtain the correct direction of causal influence.

3. Third, subsampling the processes can produce misleading Granger Causal relations [81]. Performing Granger Causal analysis on subsampled time series can lead one to miss the causal influence. If pre-processing involves subsampling, then this should be done with care.

It is important to note that the technical objections listed above are all deficiencies in or limitations of *measurement*. They indicate that incorrect Granger Causal influences may be estimated if there is some deficiency or limitation in the measurement procedure. These objections can be resolved by taking better measurements (by sampling more nodes, using sensors with higher signal-to-noise ratio, etc.).

Our objection, on the other hand, is more fundamental. *Even if* the measurements are made with infinite accuracy, and the regression coefficients associated with computing Granger Causality are precisely estimated, *and* the Granger Causality metric is perfectly computed (as is the case in our counter-examples), Granger Causal influence may *still* not yield the correct direction of information flow. To our knowledge, the argument that greater Granger causal influence can be opposite to the direction of information flow is a novel one. We believe that this argument is much more serious than previous objections, at least in the context of determining the directions of information flow in neuroscientific experiments, towards understanding the computational functions of brain regions.

A more serious objection has to do with the difference between statistical measures of causality (such as Granger Causality) and true causality, and whether or not our work simply alludes to this difference. Our principal argument is different, however, as we describe in remark 1 in Section 6.1.5.

6.1.4 Our counter-examples and our main result

This paper considers two experiments (introduced in Section 6.2) where a transmitter Tx wants to communicate a message to a receiver Rx in presence of a feedback channel (in one experiment, the feedback link is noiseless, while in the other it is noisy). We assume that the experimenter is able to record the transmissions of Tx and Rx using some probing mechanism. Provided with these measurements, the experimenter wants to estimate the direction of information flow, which in this context is the direction of flow of the message. Our results, derived in Section 6.3 and numerically illustrated in Section 6.4.1, show that the direction of information flow can be incorrectly inferred using both Granger causality and Directed Information. The first experiment considers the (unrealistic) case of communication across noiseless feedback channels. The second experiment allows for noise in the feedback channel. In both cases, linear strategies inspired by the scheme of Schalkwijk and Kailath [184] are used.

Our goal here is to bring out the point that whether Granger causality and Directed Information can be used to interpret the direction of information flow is an issue that can be, and perhaps should be, considered using thought experiments on simple communication

problems where *information flow direction, and quantity, is already known*. If the direction of causal influence yielded by Granger causality or some other similar measure were to match the known direction of information flow, then that measure can be more confidently used in experiments. While our results strongly suggest that one needs to exercise care in interpreting Granger causality and Directed Information dominance as an indicator for the direction of information flow, there are several shortcomings that need to be addressed in order to understand the issue at depth. These shortcomings are discussed in detail in Section 6.1.5.

We find it interesting to note that the mathematical machinery used in this paper amounts to routine arithmetic. Even simple counter-examples that do not employ difficult proof techniques are able to demonstrate our main result. This simplicity leads us to think that this counter-example is not very special, and that directions of stronger Granger Causal influence and information flow might have little to do with each other in more complex and/or noisy networks.

6.1.5 Possible objections to, and shortcomings of this work

A review of a previous conference submission of this paper had raised some objections to this work, which we discuss here to clarify our perspective. Further, our analysis has certain shortcomings, which we acknowledge. These will form the basis for future work.

1. Previous work has already noted that Granger Causal influence does not imply true causation [185]. This distinction is made rigorously by Judea Pearl [186], where he classifies Granger Causality as stemming from a "statistical" model, rather than from a "causal" model. The argument we make is very similar in spirit: we ask whether or not Granger Causality gives the correct direction of *information flow*. A question on the novelty of our work may therefore be raised: if our argument boils down to a restatement of Pearl's distinction, then this work has no new conceptual contribution. We make the case that our argument *is* novel in the following manner: in the systems we consider – the brain, as well as the communication system in our counter-example – causal influences exist in both directions. In our counter-example, for instance, the feedback communication algorithm that is employed involves transmissions from both Tx and Rx. The transmissions of each depend on what was transmitted by the other in the previous time instant. Causal influence and true causation, therefore, exist in both directions. The message, however, flows in only one direction: from the transmitter to the receiver. We ask whether or not the direction of *this* information flow can be discerned by comparing Granger causal influences in each direction. To this end, we give a concrete counter-example. This work is particularly relevant in the context of modern neuroscience, where such directions of information flow are desired in order to understand brain function.

2. Our analysis does not tackle an information source that evolves with time. Hence, our communication process (inspired from the scheme of Schalkwijk and Kailath [184]) is non-ergodic. Since Granger causality is really just relative errors in prediction of a process, in the presence or absence of knowledge of another process, we compute the obvious generalization of Granger causality to non-ergodic processes. Nevertheless,

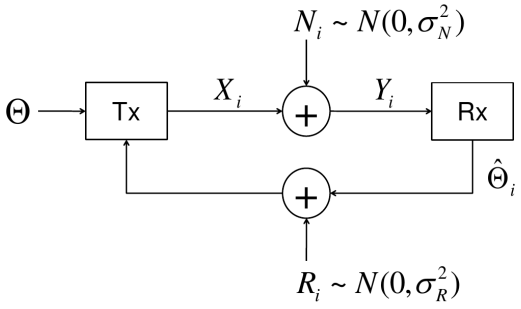


Figure 6.3: A block-diagram representation of the communication system, describing the feedback channels and supplying notation for the variables used throughout the paper. Note that this diagram is the more general of the two cases discussed in sub-sections 6.2.1 and 6.2.2, as it contains noise in the feedback link. The former, noiseless, case is equivalent to setting σ_R^2 to 0.

future work will address a situation with a linear dynamical system as the information source.

3. Our experiments restrict themselves to Gaussian noise for simplicity, but neural spiking and spike-rate models for spikes tend to be very different from those used here. This is a clear direction for future work.

4. The power and energy constraints are somewhat oversimplified to make the analysis simpler. This is for simplicity of exposition. A more general analysis is a simple extension.

5. We have also restricted ourselves to analyzing linear feedback communication strategies. In the presence of noise in the feedback link, linear communication strategies are known to be sub-optimal [187]. In order to make a water-tight argument, we would need to show that Granger Causality fails to correctly predict the direction of information flow, even when an optimal (non-linear) communication strategy is employed. This could be scope for future work. However, we do not expect results to change dramatically: when the feedback link is impaired by noise of only very low variance, the noiseless case should make for a good approximation of the system, and results should degrade gracefully, if there is any degradation at all.

6. We consider a simple point-to-point network. In general networks, this issue could be even more complex. However, since this issue shows up even for the simplest network, we feel that the problem will only be exacerbated when the network is large.

6.2 A simple feedback communication scheme

This section summarizes the analysis in the work of Schalkwijk and Kailath [184], and a simple (and previously known) extension to a noisy feedback case. It also establishes the model and the notation used in the paper.

The transmitter wants to convey a single zero-mean random number, Θ, having variance σ_θ^2, to the receiver. Θ could be obtained, for instance, by quantizing a bounded interval on

the real line (e.g. $[-1, 1]$), as is done in the scheme of Schalkwijk and Kailath [184]. The forward channel is an AWGN channel with noise variance σ_N^2. We will use simple linear communication strategies for a noiseless feedback channel, as well as an AWGN feedback channel with noise variance σ_R^2. In both cases, the estimators will be shown to be unbiased and consistent (the error mean is zero, and the error variance converges to zero).

6.2.1 Noiseless feedback: the Schalkwijk-Kailath strategy

In the first step, the transmitter sends[3] $X_1 = \Theta$, which the receiver receives with added noise. The receiver sends back an estimate of Θ over the feedback link. In all subsequent iterations, the transmitter sends the receiver the error in its latest estimate.

Therefore, in general, the transmitter sends

$$X_i = \Theta - \widehat{\Theta}_{i-1} \tag{6.1}$$

and the receiver receives

$$Y_i = X_i + N_i \tag{6.2}$$

where $N_i \sim \mathcal{N}(0, \sigma_N^2)$ iid. The receiver then estimates

$$\widehat{\Theta}_i = \widehat{\Theta}_{i-1} + \frac{Y_i}{i} \tag{6.3}$$

which results in:

$$\begin{aligned}
\widehat{\Theta}_i &= \widehat{\Theta}_{i-1} + \frac{X_i + N_i}{i} \\
&= \widehat{\Theta}_{i-1} + \frac{\Theta - \widehat{\Theta}_{i-1} + N_i}{i} \\
&= \frac{(i-1)\widehat{\Theta}_{i-1} + \Theta + N_i}{i} \\
i\widehat{\Theta}_i &= (i-1)\widehat{\Theta}_{i-1} + \Theta + N_i \\
&= (i-2)\widehat{\Theta}_{i-2} + \Theta + N_{i-1} + \Theta + N_i \\
&\vdots \\
&= i\Theta + \sum_{j=1}^{i} N_j \\
\widehat{\Theta}_i &= \Theta + \frac{1}{i}\sum_{j=1}^{i} N_j
\end{aligned} \tag{6.4}$$

Through this scheme, the estimate $\widehat{\Theta}_i$ is seen to converge to Θ in mean-square sense as $i \to \infty$:

$$\mathbb{E}[\widehat{\Theta}_i] = \mathbb{E}\left[\Theta + \frac{1}{i}\sum_{j=1}^{i} N_j\right] = \mathbb{E}[\Theta] + 0$$

[3] We assume that the power constraints are such that the scaling constant 'α' in [184] is 1.

$$\begin{aligned}\mathbb{E}[(\widehat{\Theta}_i - \Theta)^2] &= \mathbb{E}\left[\left(\frac{1}{i}\sum_{j=1}^{i} N_j\right)^2\right] \\ &= \frac{1}{i^2}\mathbb{E}\left[\left(\sum_{j=1}^{i} N_j\right)^2\right] \\ &= \frac{i}{i^2}\mathbb{E}[N_1^2] = \frac{\sigma_N^2}{i} \overset{i\to\infty}{\longrightarrow} 0.\end{aligned}$$

6.2.2 Noisy feedback

In the presence of noise in the feedback link, restricting our attention to linear strategies, we can use a simple modification of the Schalkwijk-Kailath strategy, incorporating the feedback[4]. The receiver still simply transmits the estimate $\widehat{\Theta}_i$ based on the i-th forward channel output $Y_i = X_i + N_i$. The transmitter now receives corrupted versions $Z_i = \widehat{\Theta}_{i-1} + R_{i-1}$ of the receiver's transmissions. That is,

$$\text{Transmitter's transmissions: } X_i = \Theta - (\widehat{\Theta}_{i-1} + R_{i-1}) \tag{6.5}$$

$$\text{Channel outputs at the receiver: } Y_i = X_i + N_i \tag{6.6}$$

$$\text{Receiver's estimates \& transmissions: } \widehat{\Theta}_i = \widehat{\Theta}_{i-1} + \frac{Y_i}{i} \tag{6.7}$$

where R_{i-1} is the AWGN noise in the reverse link. $R_i \sim \mathcal{N}(0, \sigma_R^2)$ are iid. random variables.

Linear strategies are known to be suboptimal for this communication problem [187] (where Θ is a quantized random variable communicating a finite-rate message reliably), and for problems with non-classical information structures in general [188]. Nevertheless, we now show that the resulting estimates $\widehat{\Theta}_i$ still converge to Θ in mean-square sense:

$$\begin{aligned} i\widehat{\Theta}_i &= i\widehat{\Theta}_{i-1} + X_i + N_i \\ &= i\widehat{\Theta}_{i-1} + \Theta - \widehat{\Theta}_{i-1} - R_{i-1} + N_i \\ &= (i-1)\widehat{\Theta}_{i-1} + \Theta - R_{i-1} + N_i \\ &\overset{(a)}{=} i\Theta + \sum_{k=1}^{i} N_k - \sum_{k=1}^{i-1} R_k \end{aligned} \tag{6.8}$$

$$\widehat{\Theta}_i = \Theta + \frac{1}{i}\sum_{k=1}^{i} N_k - \frac{1}{i}\sum_{k=1}^{i-1} R_k \tag{6.9}$$

where (a) is obtained by expanding $\widehat{\Theta}_j$ recursively for $j = i-1, i-2, \ldots, 2$. Therefore, the error in estimating Θ converges to 0 in mean-square sense (*i.e.*, $\widehat{\Theta}_i$ is a consistent estimate of Θ_i even in the presence of noise on the feedback link).

[4] We note that implicitly, this strategy assumes an energy/SNR constraint on the feedback link. This is because the receiver simply sends back the estimate, which is shown to converge to the true value of Θ.

6.3 Granger causality and directed information analyses for the strategies in Section 6.2

6.3.1 Granger causality for the noiseless feedback case

In order to compute the Granger causality in the reverse direction (from the receiver to the transmitter), we model X_i as a linear function of its past.

$$X_i = \sum_{j=1}^{p} \alpha_j X_{i-j} + \epsilon_i \tag{6.10}$$

We then compute coefficients α_j such that the average error in fitting X_i is minimized. Note that α_j can themselves depend on i, since this is a non-stationary process (because the error $\Theta - \widehat{\Theta}_i = X_i$ converges to zero). We describe how these coefficients might be estimated in a more general setting, and justify using theoretically determined system parameters as regression coefficients in section 6.3.4.

For now, we theoretically evaluate the system parameters. We start with equation (6.1) and manipulate terms to arrive at an equation bearing the required form of equation (6.10):

$$\begin{aligned} X_i &= \Theta - \widehat{\Theta}_{i-1} \\ &= \Theta - (\widehat{\Theta}_{i-2} + \frac{Y_{i-1}}{i-1}) \\ &= \Theta - (\Theta - X_{i-1}) - \frac{X_{i-1} + N_{i-1}}{i-1} \\ &= X_{i-1} - \frac{X_{i-1} + N_{i-1}}{i-1} \\ &= \frac{i-2}{i-1} X_{i-1} + \frac{N_{i-1}}{i-1} \end{aligned}$$

Therefore, $\alpha_1 = \frac{i-2}{i-1}$ and $\epsilon_i = \frac{N_{i-1}}{i-1}$, and hence $\mathrm{Var}(\epsilon_i) = \frac{\sigma_N^2}{(i-1)^2}$.

Next, we model X_i in terms of both the past of X and the past of $\widehat{\Theta}$:

$$X_i = \sum_{j=1}^{p} \alpha_j X_{i-j} + \sum_{j=1}^{p} \beta_j \widehat{\Theta}_{i-j} + \tilde{\epsilon}_i \tag{6.11}$$

We can manipulate equation (6.1) to bring it into the above form:

$$\begin{aligned} X_i &= \Theta - \widehat{\Theta}_{i-1} \\ &= X_{i-1} + \widehat{\Theta}_{i-2} - \widehat{\Theta}_{i-1} \end{aligned}$$

Since there is no noise expression here, the Granger causality ratio, $\mathrm{Var}(\epsilon_i)/\mathrm{Var}(\tilde{\epsilon}_i)$ goes to infinity.

In the forward direction, we do not explicitly compute the Granger causality ratio, but simply show that it is bounded strictly between 1 and ∞:

$$\widehat{\Theta}_i = \sum_{j=1}^{p} \alpha_j \widehat{\Theta}_{i-j} + \epsilon_i \tag{6.12}$$

$$\begin{aligned}\widehat{\Theta}_i &= \widehat{\Theta}_{i-1} + \frac{X_i + N_i}{i} \\ &= \widehat{\Theta}_{i-1} + \frac{\Theta - \widehat{\Theta}_{i-1} + N_i}{i} \\ &= \frac{i-1}{i}\widehat{\Theta}_{i-1} + \frac{\Theta}{i} + \frac{N_i}{i}\end{aligned}$$

which means that $0 < \frac{\sigma_N^2}{i^2} < \mathrm{Var}(\epsilon_i) < \frac{\sigma_N^2+\sigma_\theta^2}{i^2} < \infty$, since the past of $\widehat{\Theta}$ cannot be used to explain N_i. Further, if we try to predict $\widehat{\Theta}_i$ from the previous $\widehat{\Theta}_{i-j}$ and X_{i-j}:

$$\widehat{\Theta}_i = \sum_{j=1}^{p} \alpha_j \widehat{\Theta}_{i-j} + \sum_{j=0}^{p-1} \beta_j X_{i-j} + \tilde{\epsilon}_i \tag{6.13}$$

we see that

$$\widehat{\Theta}_i = \widehat{\Theta}_{i-1} + \frac{X_i}{i} + \frac{N_i}{i}$$

so that $\mathrm{Var}(\tilde{\epsilon}_i) = \sigma_N^2/i^2$. The Granger causality ratio, $\mathrm{Var}(\epsilon_i)/\mathrm{Var}(\tilde{\epsilon}_i)$, in the forward direction is, therefore, finite.

The intuitive argument for why this is happening might go as follows: since the feedback link is noiseless, one can always perfectly predict the transmitted symbol from the past $\widehat{\Theta}$'s and the history of X. On the other hand, one can never perfectly predict $\widehat{\Theta}_i$ from the past X's and the history of $\widehat{\Theta}$.

6.3.2 Directed Information for the noiseless feedback case

Performing the Directed Information analysis for the scheme described above yields the same results. In order to ease the burden of computing Directed Information, we assume that Θ is normally distributed.

The directed information in the forward direction is computed as:

$$I(X^n \to \widehat{\Theta}^n) = \frac{1}{2}\log\left(1 + \frac{n\sigma_\theta^2}{\sigma_N^2}\right)$$

where n is the number of iterations of the Schalkwijk-Kailath algorithm. For a proof, refer Appendix 6.A.

In the reverse direction, the Directed Information is ∞.

$$\begin{aligned}I(0 * \widehat{\Theta}^{n-1} \to X^n) &= \sum_{i=0}^{n-1} I(X_{i+1}; \widehat{\Theta}^i | X^i) \\ I(X_{i+1}; \widehat{\Theta}^i | X^i) &= h(X_{i+1}|X^i) - h(X_{i+1}|X^i, \widehat{\Theta}^i)\end{aligned} \tag{6.14}$$

The first term in the equation above reduces to

$$\begin{aligned}h(X_{i+1}|X^i) &= h(\Theta - \widehat{\Theta}_i | X^i) \\ &= h\left(\Theta - \left(\widehat{\Theta}_{i-1} + \frac{X_i + N_i}{i}\right)\middle| X^i\right)\end{aligned}$$

$$\begin{aligned}
&\overset{(a)}{=} h\left(\Theta - \left((\Theta - X_i) + \frac{N_i}{i}\right)\middle| X^i\right) \\
&= h\left(\frac{N_i}{i}\middle| X^i\right) \\
&= h(N_i) - \log(i) \\
&= \frac{1}{2}\log(2\pi e\sigma_N^2) - \log(i)
\end{aligned}$$

where for (a), we have dropped X_i/i from the previous step, since it is conditioned over, and then written $\widehat{\Theta}_{i-1}$ as $(\Theta - X_i)$. On the other hand, the second term in equation (6.14) becomes

$$\begin{aligned}
h(X_{i+1}|X^i, \widehat{\Theta}^i) &= h(\Theta - \widehat{\Theta}_i|X^i, \widehat{\Theta}^i) \\
&= h(\Theta|X^i, \widehat{\Theta}^i) \\
&\overset{(a)}{=} h(X_i + \widehat{\Theta}_{i-1}|X^i, \widehat{\Theta}^i) \\
&= h(0|X^i, \widehat{\Theta}^i) \\
&\overset{(b)}{=} -\infty
\end{aligned}$$

where for (a) we have expressed Θ in terms of $\widehat{\Theta}_{i-1}$ and X_i and for (b), we have used the fact that the differential entropy of a constant (or equivalently, a Gaussian with zero variance) is negative infinity. This means that equation (6.14) becomes

$$\begin{aligned}
I(X_{i+1}; \widehat{\Theta}^i|X^i) &= \infty \\
\Rightarrow I(0 * \widehat{\Theta}^{n-1} \to X^n) &= \infty
\end{aligned}$$

6.3.3 Directed Information for the noisy feedback scenario

Since a noiseless feedback link is not realistic, we proceed to perform the same Directed Information calculations as above for the feedback link with additive white Gaussian noise of variance σ_R^2. While we could not derive simple closed form expressions for the Directed Information in the forward and reverse links, we were able to evaluate the expressions numerically. These are plotted in section 6.4.1.

The Directed Information in the forward direction can be written as

$$\begin{aligned}
I(X^n \to \widehat{\Theta}^n) &= \sum_{i=1}^{n} I(\widehat{\Theta}_i; X^i|\widehat{\Theta}^{i-1}) \\
&= \sum_{i=1}^{n} h(\widehat{\Theta}_i|\widehat{\Theta}^{i-1}) - h(\widehat{\Theta}_i|\widehat{\Theta}^{i-1}, X^i) \qquad (6.15) \\
&= \sum_{i=1}^{n} \left(\frac{1}{2}\log(2\pi e\mathrm{Var}[\Theta - R_{i-1} + N_i|\widehat{\Theta}^{i-1}]) \right. \\
&\qquad \left. - \frac{1}{2}\log(2\pi e\sigma_N^2)\right)
\end{aligned}$$

For a derivation of this, see Appendix 6.B. In the reverse direction,

$$\begin{aligned} I(0 * \widehat{\Theta}^{n-1} \to X^n) &= \sum_{i=0}^{n-1} I(X_{i+1}; \widehat{\Theta}^i | X^i) \\ &= \sum_{i=0}^{n-1} h(X_{i+1}|X^i) - h(X_{i+1}|X^i, \widehat{\Theta}^i) \\ &= \frac{1}{2} \log \left(2\pi e \frac{\sigma_N^2 + \sigma_R^2}{\sigma_R^2} \right) \\ &+ \sum_{i=2}^{n-1} \left(\frac{1}{2} \log \left(2\pi e \mathrm{Var} \left[R_{i-1} - \frac{N_i}{i} - R_i \middle| X^i \right] \right) \right. \\ &\qquad \left. - \frac{1}{2} \log(2\pi e \sigma_R^2) \right) \end{aligned} \tag{6.16}$$

For a derivation of this, see Appendix 6.C.

6.3.4 A note on estimating regression coefficients

The Granger Causality and Directed Information metrics are a function of the regression coefficients estimated from the data by fitting the models given by equations (6.10) and (6.11). In our analysis, we described the data-generation model: the algorithm inspired by the Schalkwijk and Kailath scheme for feedback communication. We then proceeded to use the system parameters of this model directly as regression coefficients in our Granger Causality computation. This could be construed as being erroneous: we ought to simulate the data generation, and estimate the regression coefficients from the generated data. This would better model the actions of the neuroscientist who seeks to perform Granger Causality analysis.

We justify our knowledge of the system parameters and their use as regression coefficients in the following manner: we assume that the regression coefficients can be accurately estimated from data, since neuroscientific experiments typically record the *same* processes multiple times – these are called "trials". The availability of multiple trials of the same process can be leveraged to estimate the system parameters accurately, even if the processes are non-stationary.

Suppose we record two non-stationary processes, $\{X_t\}_{t=1}^n$ and $\{Y_t\}_{t=1}^n$, for which we seek to compute the Granger Causality metrics. To this end, we must find coefficients $\alpha_j(t)$ and $\beta_j(t)$ to minimize the error in fitting the models given by equations (6.10) and (6.11). Note that α and β depend on t, since we assume the process is non-stationary. However, since we record the *same* process in each trial, the $\alpha_j(t)$ and $\beta_j(t)$ are constant across trials. Estimating them from data that has many trials is then a simple matter of linear regression.

For a given time instant t, the i^{th} trial is modeled as

$$X_t^{(i)} = \sum_{j=1}^{p} \alpha_j(t) X_{t-j}^{(i)} + \epsilon_t^{(i)}$$

Note that $\alpha_j(t)$ does not depend on i. Collecting variables across N trials, we can write the full model in vector form:

$$\begin{bmatrix} X_t^{(1)} \\ X_t^{(2)} \\ \vdots \\ X_t^{(N)} \end{bmatrix} = \begin{bmatrix} X_{t-1}^{(1)} & \cdots & X_{t-p}^{(1)} \\ X_{t-1}^{(2)} & \cdots & X_{t-p}^{(2)} \\ \vdots & \ddots & \vdots \\ X_{t-1}^{(N)} & \cdots & X_{t-p}^{(N)} \end{bmatrix} \begin{bmatrix} \alpha_1(t) \\ \alpha_2(t) \\ \vdots \\ \alpha_p(t) \end{bmatrix} + \begin{bmatrix} \epsilon_t^{(1)} \\ \epsilon_t^{(2)} \\ \vdots \\ \epsilon_t^{(N)} \end{bmatrix}$$

If we call the vector on the LHS $\mathbf{Y}$ and the matrix on the RHS $\mathbf{X}$, then the vector of $\alpha_j(t)$'s ($\alpha(t)$) can be estimated at time instant t using Ordinary Least Squares:

$$\hat{\alpha}(t) = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}$$

This is an unbiased and consistent estimator for $\alpha(t)$. This analysis can be trivially extended to the model described by equation (6.11). With a sufficiently large number of trials, therefore, the system parameters (to be used as regression coefficients in the Granger Causality analysis) can be estimated to arbitrarily high accuracy.

It should be noted that we have restricted ourselves to an analysis of linear strategies: the channel, the extended Schalkwijk and Kailath scheme to a noisy feedback link, and the proposed regression model are all linear.

As a final remark, we note that estimating the regression coefficients accurately is a conservative assumption on our part. As mentioned at the end of section 6.1.3, we see that *despite* being computed accurately, the metrics of Granger Causality and Directed Information incorrectly estimate the direction of information flow. A rigorous analysis would warrant the computation of these coefficients in simulation, for a finite number of trials. It is our belief, however, that our result is unlikely to degrade if the coefficients are not estimated perfectly. Future work will address this matter in greater depth.

6.4 Analytical and Simulation Results

6.4.1 Numerical results on the direction of greater Granger causal influence

In the noiseless feedback case, the directed information on the forward link is finite, while that on the feedback link is infinite (refer sections 6.3.1 and 6.3.2). However, for completeness, we examine the case when noise is present in the feedback link, as is illustrated in Fig. 6.4, through numerical calculation of expressions in the last section. For cases where the noise variance of the feedback link (σ_R^2) is moderately smaller than feedforward noise variance (σ_N^2), we observe that the Directed Information in the direction of the reverse link can dominate that in the direction of the forward link. With sufficiently many (albeit sometimes a large number of) iterations, the Directed Information in forward direction starts to dominate. However, the point at which this happens depends on the (often unknown) ratio of noise in these links.

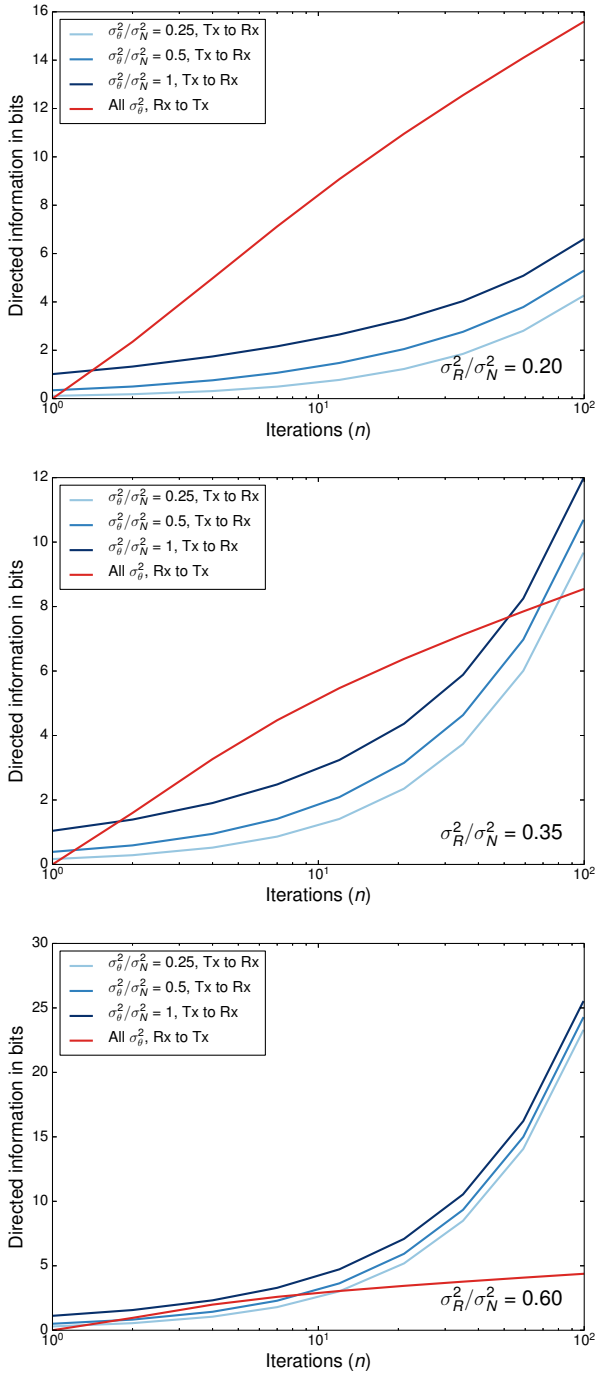


Figure 6.4: Plots for forward and backward directed information computations for $\sigma_R^2/\sigma_N^2 = 0.2$ (top), 0.35 (center) and 0.6 (bottom). In each plot, curves for directed information in both directions are illustrated for ratios $\sigma_\theta^2/\sigma_N^2 = 0.25, 0.5$, and 1. The x-axis is the number of iterations of message-passing between the transmitter and the receiver. For cases when feedback noise variance σ_R^2 is moderately smaller than feedforward noise variance σ_N^2, directed information in the reverse link can dominate that in the forward link. With sufficiently many (albeit sometimes large, as illustrated in the top figure) iterations, directed information in forward information starts to dominate. However, the point at which this happens depends on the (often unknown) ratio of noise in these links.

6.4.2 Simulations showing statistically insignificant GCI in the direction of information flow

So far, we have only considered whether the direction of *greater* Granger causal influence can be opposite to the direction of information flow. We used numerical methods to approximate the analytical expressions for directed information (a generalization of Granger causal influence). As a result, questions on the statistical significance of Granger causal influence never arose. In this section, we consider whether the Granger causal influence can be *statistically insignificant in the direction of information flow* while being *statistically significant in the opposite direction*. This would imply a much stronger result that what we have considered so far: it would mean that, in the presence of feedback networks, it is possible that a Granger causal analysis mistakes certain information flows to be the a direction precisely opposite to their actual nature.

We demonstrate this computationally in Fig. 6.5. We simulated the Schalkwijk and Kailath scheme for $T = 100$ time steps and for $n = 100$ trials. We computed GCIs by fitting an autoregressive model of order $p = 10$ to the data. Fig. 6.5 shows the mean GCI over 100 trials (errorbars represent standard error of the mean). We assessed the statistical significance of the result using the method described by [10]: we permuted the trials of the transmitter's and receiver's transmissions independently, to disrupt trial-related dependences, while maintaining the original distributions of the individual transmissions. We then computed the GCIs on the permuted trials. We repeated this process $n_{\text{Perm}} = 100$ times, and constructed a histogram of mean GCIs under permutation, which became our empirical estimate of the null distribution. We found that for a certain regime of σ_R/σ_N, the actual GCI from the receiver to the transmitter was far outside the empirical null distribution. The p-value of 0.01 was effectively the minimum attainable p-value, determined by the number of permutations we performed. Fig. 6.5 shows that GCIs can be statistically insignificant in the direction of information flow (from X to $\widehat{\Theta}$), while at the same time being highly significant in the opposite direction (from $\widehat{\Theta}$ to X).

6.5 Resolving the Counterexample using M-Information Flow

The Schalkwijk and Kailath scheme [189] is an efficient strategy for communicating a message in the presence of a noisy feedforward channel and a noiseless feedback channel. We have previously used this scheme as a counterexample [82], to show that comparing Granger causal influences in forward and backward directions can lead to erroneous inferences on the direction in which the message is being sent in this feedback system. We first provide a brief overview of the scheme, then recapitulate our previous result, and finally demonstrate what the information flow framework developed in this paper has to offer in the case of this example.

Consider the communication system depicted in Figure 6.6, which shows the schematic of a simplified version of the Schalkwijk and Kailath scheme. For convenience, let us denote the transmitter, A, and receiver, B, by Alice and Bob respectively. Alice is attempting to communicate a message M to Bob over an additive Gaussian channel, but in the presence

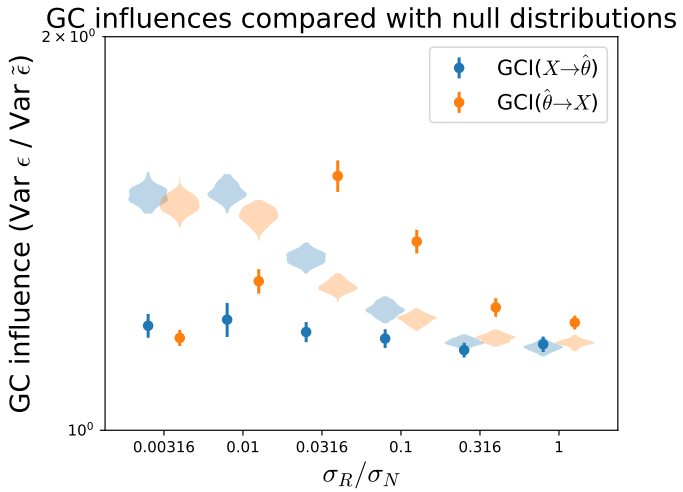


Figure 6.5: A comparison of Granger Causal influences (GCIs) at different reverse-noise-ratios, σ_R/σ_N. The violin plots indicate the null distributions based on the permutation test described in Section 6.4.2, while the errorbars show the mean and standard error of GCI. $\sigma_N = 0.1$ for this plot.

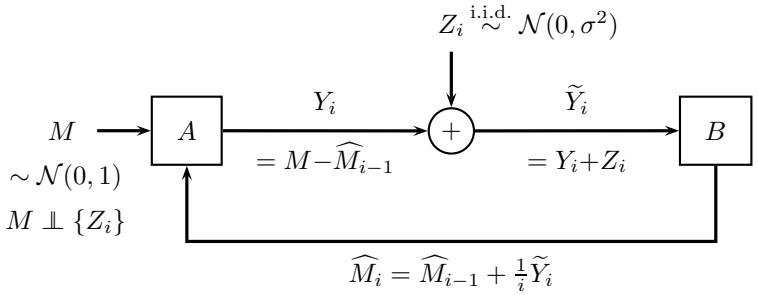


Figure 6.6: A communication system depicting the Schalkwijk and Kailath scheme. Alice, represented by node A, communicates a message M to Bob, represented by node B, in the presence of a noisy feedforward channel and a noiseless feedback channel. In the i^{th} iteration, Alice transmits the error in Bob's most recent estimate of the message, Y_i, but her transmission is corrupted by the noise Z_i. Bob updates and transmits his estimate, $\widehat{M}_i$, which reach Alice noiselessly.

of noiseless feedback. Alice starts by transmitting the message $Y_1 = M$ to Bob, over the noisy feedforward channel. Bob receives a corrupted version of M, given by $\widetilde{Y}_1 = Y_1 + Z_1$, and computes an estimate $\widehat{M}_1$. He sends this estimate back to Alice over the noiseless feedback channel. In the iterations that follow, Alice computes the error in Bob's most recent estimate, $Y_i = M - \widehat{M}_{i-1}$, and sends this to Bob over the noisy feedforward channel. Meanwhile, Bob updates his estimate based on Alice's noisy transmissions $\widetilde{Y}_i = Y_i + Z_i$, using the following rule:

$$\widehat{M}_i = \widehat{M}_{i-1} + \frac{1}{i}\widetilde{Y}_i \tag{6.17}$$

It can be shown that this rule implies

$$\widehat{M}_i = M + \frac{1}{i}\sum_{j=1}^{i} Z_j \tag{6.18}$$

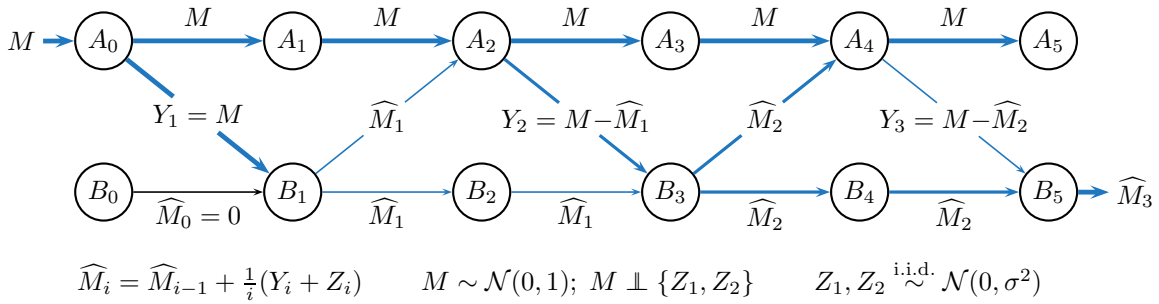


Figure 6.7: A computational system describing the first few iterations of the Schalkwijk and Kailath scheme. Almost every edge shown here has M-information flow. However, the *quantity* of M-information flow (shown using line thickness) reveals the asymmetry between Alice and Bob: Alice has the message to begin with, and her transmissions have a larger volume of M-information flow. In contrast, Bob's initial transmissions are poor estimates and have small volumes of M-information flow, but they get better over a few iterations, and eventually come close to the true message. Furthermore, we also reveal an asymmetry between Alice and Bob using the concept of derived information: each of Bob's transmissions is M-derived from Alice's previous transmissions, whereas Alice's transmissions are *not* M-derived from Bob's previous transmissions. Both these facts point towards the idea that Alice is slowly sending information about M to Bob.

Thus, this strategy ensures that Bob's estimate $\widehat{M}_i$ converges to M in mean squared sense [82].

Intuitively, one might expect that, since the message M is being transmitted in the forward direction, the Granger causal influence from Alice to Bob is greater than that from Bob to Alice. However, our earlier result [82] showed that, in fact, the opposite is true. In other words, even though the message is being communicated from Alice to Bob, the Granger causal influence from Bob to Alice is greater; in fact, the Granger causal index from Bob to Alice is *infinite*. The reason for this is that, while Alice's past transmissions do not perfectly predict Bob's transmissions (due to the presence of noise in the feedforward link), Bob's past transmissions *perfectly* predict Alice's transmissions (since the latter are a simple function of the former). Therefore, the Granger causal index from Alice to Bob, which measures the relative predictive gain of including Alice's past transmissions in the autoregression for Bob's transmissions, remains finite; while the Granger causal index from Bob to Alice becomes infinite.

Our earlier paper on this subject [82] concluded that the direction of greater Granger causal influence could be opposite to the "direction of information flow" in the Schalkwijk and Kailath scheme. There, "information flow" was being used purely in an intuitive sense, to mean the direction in which the message was being communicated in that system. The intent of our previous paper was to explain that it is not always possible to interpret a larger Granger causal influence in a certain direction to mean that a specific message is being communicated in that direction. In contrast, this paper presents a refined theoretical framework that defines information flow about a message M for a specific *edge* in a computational system. Now, we no longer speak of *one specific direction* in which information flows; rather, we describe *which edges* carry information about the message in their transmissions *at each point in time*. This leads to a more nuanced understanding of information flow in the Schalkwijk-Kailath setting.

Before we can analyze the M-information flows in the Schalkwijk-Kailath scheme, we need to fit the scheme within the computational system framework. Figure 6.7 shows

the time-unrolled computational system corresponding to two feedforward and feedback iterations of the simplified Schalkwijk-Kailath scheme described before. In order to translate the communication system into our computational system model while remaining consistent with our earlier work [82], we have merged the process of noise addition with the receiver, i.e., Bob. This exposes the edges with Alice's and Bob's *transmissions*, making them observable, as was assumed in our previous paper [82]. This is also consistent with what *would have been* observable if A and B were neurons (or neural populations) whose outputs a neuroscientist were to measure.[5] Note that one full iteration of the Schalkwijk-Kailath scheme takes two time steps in this model, so the iteration index i advances once for every two time steps t. Also, note that merging noise-addition with the receiver does *not* make $\widetilde{Y}$ or Z "hidden nodes", since the function computed at B_t can be defined purely in terms of its inputs, $(Y_i, \widehat{M}_{i-1})$, and its intrinsic random variable, $W(B_t)$ (which absorbs Z_i), as follows:

$$f_{B_{2i-1}}(Y_i, \widehat{M}_{i-1}, W(B_{2i-1})) = \widehat{M}_{i-1} + \frac{1}{i}(Y_i + W(B_{2i-1})) \tag{6.19}$$

where $W(B_{2i-1}) = Z_i$ takes the role of the noise in the communication system. Also, to understand the time index for node B, note that in the first step of iteration i, Alice transmits to Bob, i.e., node A_{2i-2} transmits to B_{2i-1} (see Figure 6.7).

Now, we first show that all edges depicted in blue in Figure 6.7 carry M-information flow, based on Definition 2.4. Specifically, both Alice's feedforward transmissions *and* Bob's feedback transmissions have M-information flow. This should not be surprising for the following intuitive reasons: Alice's transmissions convey information about M which Bob uses to improve his estimate; meanwhile, Bob's transmissions are estimates of M, and therefore must depend on M.

In fact, we can take this intuitive argument further: suppose we were to *quantify* M-information flow by using the following natural extension of our definition,

$$\mathcal{F}_M(E_t) \coloneqq \max_{\mathscr{E}'_t \subseteq \mathscr{E}_t} I\big(M; X(E_t) \,\big|\, X(\mathscr{E}'_t)\big). \tag{6.20}$$

Noting that Definition 2.4 only specified *whether or not* a given edge E_t had information flow, all that we have now done is to take the maximum over the subsets of edges used to discover M-information flow in that definition. This quantification is fully consistent with our definition of M-information flow, since it goes to zero if and only if the M-information flow on an edge goes to zero. Now, using this quantitative notion of information flow, we can ask how the M-information flow on a given link—feedforward or feedback—varies with time. In particular, it should be intuitively clear that the M-information content in Bob's transmissions, i.e. $\widehat{M}_i$, *increases* over time as his estimate improves. This is depicted as an increase in the thickness of the edges carrying Bob's transmissions with time.

On the other hand, the information content in Alice's transmissions *decreases* with time. To understand why this is true, first note that $I(M; Y_i) = 0$ for $i > 1$, since Y_i carries only information about the noise in $\widehat{M}_{i-1}$ (after the first iteration), which is independent of M, as seen in Equation (6.18). However, since Alice's transmissions represent the *noise* in

[5] From a wireless communication system perspective, as well, it is more reasonable to assume noise to be a part of the receiver's node, since the additive noise in a signal is usually considered to be the result of thermal noise in the receiver's circuitry.

Bob's estimates of M, they depend on the message when *conditioned* on Bob's estimates (this is similar to how Z carries information about M when conditioned on $M \oplus Z$ in Counterexample 2.1). So the quantified M-information flow of Alice's transmissions will be given by:

$$I(M; Y_i \,|\, \widehat{M}_{i-1}) = I(M; M - \widehat{M}_{i-1} \,|\, \widehat{M}_{i-1}) \tag{6.21}$$

$$\overset{(a)}{=} H(M \,|\, \widehat{M}_{i-1}) + H(M \,|\, M - \widehat{M}_{i-1}, \widehat{M}_{i-1}) \tag{6.22}$$

$$= H(M \,|\, \widehat{M}_{i-1}) \tag{6.23}$$

$$= H(M) - I(M; \widehat{M}_{i-1}) \tag{6.24}$$

where in (a), the second term goes to zero because M is a constant when given $\widehat{M}_{i-1}$ and $M - \widehat{M}_{i-1}$. During the initial iterations, when Bob's estimate is poor, we must have that $I(M; \widehat{M}_{i-1})$ is very small (as we might expect if the noise is large, for instance). Hence, for the first few iterations, the quantified M-information flow of Alice's transmissions will be close to $H(M)$, from Equation (6.24). However, as Bob's estimate improves, $I(M; \widehat{M}_{i-1})$ becomes closer to $H(M)$, and therefore $I(M; Y_i \,|\, \widehat{M}_{i-1})$ becomes close to zero. Thus, the quantified M-information flow of Alice's transmissions decreases over time. Correspondingly, this is depicted using edges whose thickness decreases over time in Figure 6.7. Quantifying the M-information flows of the feedforward and feedback links thus reveals an asymmetry between Alice and Bob that strongly suggests that the message is being transmitted from Alice to Bob.

We can also get a more nuanced understanding of information flow in this system by asking whether Bob's transmissions are *derived* from Alice's, or vice versa. First, consider whether Bob's transmissions are derived M-information of Alice's previous transmissions: this can be expressed in terms of the Markov chain M—$[M,\ M - \widehat{M}_1]$—$\widehat{M}_2$. Observe that this Markov chain holds trivially:

$$I(M; \widehat{M}_2 \,|\, M,\ M - \widehat{M}_1) = 0. \tag{6.25}$$

However, if we consider whether Alice's transmissions are derived M-information of Bob's past transmissions, it can be shown that M—$[\widehat{M}_1,\ \widehat{M}_2]$—$(M - \widehat{M}_2)$ is not a valid Markov chain (see Appendix 6.D for a detailed derivation). Hence, we see that Bob's transmissions are derived M-information of all of Alice's past transmissions, however, Alice's transmissions are *not* derived M-information of all of Bob's past transmissions. In conjunction with the fact that the volume of M-information flow in Alice's transmissions slowly decreases from $H(M)$ with time, while the volume of M-information flow in Bob's transmissions slowly increases to $H(M)$ with time, this suggests that Alice has some information about the message M that Bob slowly receives from Alice.

This example shows how a measure that quantifies information flow, along with derived information, can be used to understand some finer computational structure present within the computational system. In general, however, care needs to be exercised in applying derived M-information: one must choose what Markov condition to check in a principled manner. In the specific case of the Schalkwijk-Kailath example, we had the advantage of being in a two-node setting, where the derived information expressions we examined had clear interpretations. It may be that analyzing information flow first, to understand

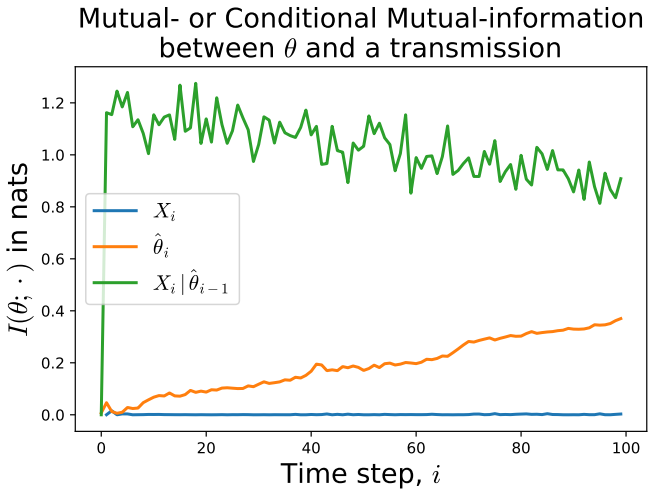


Figure 6.8: Mutual (and conditional mutual) information between the stimulus and Alice's and Bob's transmissions. $I(\theta; \hat{\theta}_i)$ slowly increases with i, while $I(\theta; X_i \mid \hat{\theta}_{i-1})$ slowly falls, indicating that Alice is communicating information about the message θ to Bob.

which variables transmit information about M to one another, can help guide the choice of variables to examine when applying derived M-information.

6.5.1 Measuring M-information flow in the simulation from Section 6.4.2

Granger causality's failure to identify the direction in which the message flows in the above example can be attributed to the fact that Granger causality only examines predictive influence; it does not capture what that influence is *about*. Granger causality does not intrinsically check for stimulus-dependence in any way. The recent work of [49], while defining stimulus-related information flow, does not provide a quantitative measure of information flow, and their partial resolution to the counterexample based on derived information is cumbersome and unsatisfactory.

Here, we take a much simpler approach and show that by measuring mutual and conditional mutual information, we can observe how information about the message evolves in Alice's and Bob's transmissions. Since all variables in this example are Gaussian, the mutual information between the message θ and any transmission U can be written in terms of their correlation: $I(\theta; U) = -\frac{1}{2}\log(1 - \rho(\theta, U)^2)$, where the correlation $\rho(\theta, U)$ is readily estimated. Fig. 6.8 shows how the mutual (and conditional mutual) information of X_i and $\hat{\theta}_i$ evolve over time steps, i. In particular, observe that $I(\theta; \hat{\theta}_i)$ slowly increases over time i, while $I(\theta; X_i)$ is nearly zero. The conditional mutual information $I(\theta; X_i \mid \hat{\theta}_{i-1})$, however, is much larger and slowly decreases over time, indicating the presence of synergistic information about θ in the forward link, which decays as the estimate $\hat{\theta}$ improves.

The decrease of stimulus-related information in Alice's transmissions, and the corresponding increase in Bob's transmissions indicates that information about the stimulus is being conveyed from Alice to Bob and not vice versa. This also indicates that caution

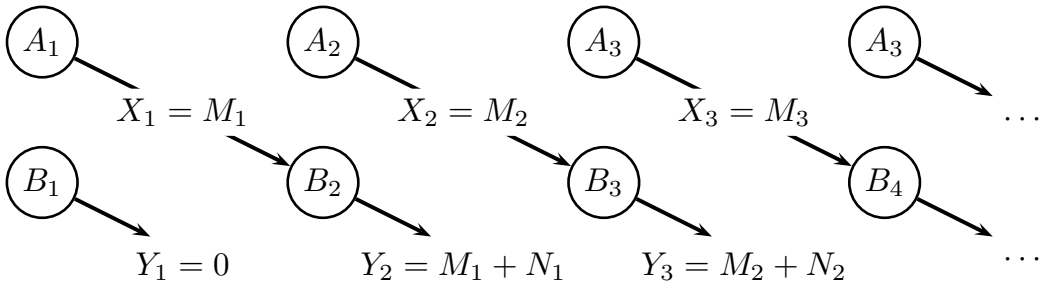


Figure 6.9: A simple example where Granger causal influence is closely related to M-information flow

must be exercised in interpreting Granger causal influences as conveying stimulus-related information.

6.6 When does Granger Causality give us M-Information Flow?

This is a question that we do not yet have a full answer for. However, we can find sufficient conditions under which Granger Causality does indeed measure a quantity that is equivalent to M-information flow. Likewise, we can provide a few examples where Granger Causality cannot work by virtue of what it measures. In this section, we also review a well-known work that provides assumptions under which Directed Information is guaranteed to correctly recovers the correct directions of all influences.

Firstly, it is straightforward to see that in a two-node feedforward example, where the message M is what is being communicated by the transmitter, Granger Causality correctly captures the direction and magnitude of information flow: Suppose that we have a message composed of many independent random variables over time: $M = [M_1, M_2, \ldots]$, where we take $M_t \sim$ i.i.d. $\mathcal{N}(0, \sigma_M^2)$ for simplicity. Further suppose that these messages are being communicated from the process $\{X_t\}$ to the process $\{Y_t\}$, as follows:

$$X_t = M_t \tag{6.26}$$

$$Y_{t+1} = X_t + N_t = M_t + N_t, \tag{6.27}$$

where $N_t \sim$ i.i.d. $\mathcal{N}(0, \sigma_N^2)$ are noise variables, which are all independent of the messages M_t. This is depicted in Figure 6.9. Then, over n time steps, we must have

$$I(X^n \to Y^n) = \sum_{t=1}^{n} I(X^t; Y_{t+1} \,|\, Y^t) \tag{6.28}$$

$$= \sum_{t=1}^{n} I\Big(\{M_s\}_{s=1}^{t}; M_t + N_t \,\Big|\, \{M_s + N_s\}_{s=1}^{t-1}\Big) \tag{6.29}$$

$$= \sum_{t=1}^{n} I(M_t; M_t + N_t) = \frac{n}{2} \log\Big(1 + \frac{\sigma_M^2}{\sigma_N^2}\Big) \tag{6.30}$$

where in the penultimate step, we can drop the conditioning, as well as all variables in $\{M_s\}_{s=1}^{t}$ except M_t, because M_t and N_t are independent of all M_s and N_s for $s < t$. If we

now consider the *sum* of M-information flows at the outgoing edges of B_t, we find:

$$\mathcal{F}_M(B^n) = \sum_{t=1}^{n} I(M; Y_t) = \frac{n}{2} \log\left(1 + \frac{\sigma_M^2}{\sigma_N^2}\right). \tag{6.31}$$

This is because there is clearly nothing we can condition upon to increase the M-information flow in this example. Therefore, in this very simple instance, the Granger causal influence is equal to the net M-information flow volume over n time instants.

It is worth noting that although we computed the Granger causal influence from A to B, the M-information flow is actually computed at the outgoing edges of B, not of A. This is because Granger Causality does not refer to information about any specific message M, rather, it refers to the influence that the process $\{X_t\}$ has on the process $\{Y_t\}$. In order to make the two measures compatible with each other, we have here taken the process $\{X_t\}$ to *itself* be the message M. Therefore, it is only natural that the Granger causal influence from A to B would be equal to the M-information flow at the outgoing edges of B.

If we wanted to undertake a more thorough analysis of this nature, but where Granger Causality refers to some specific message M, then we will need to go into exactly how the Granger causal influence is being *tested* to be dependent on M However, such an analysis is beyond the scope of this work. It is worth noting again that because Granger Causality does not intrinsically, comparing it with M-information flow would require some manner of inference that checks how much of the Granger causal influence can be attributed to M. This direction has started to receive attention only recently, once again using measures from partial information decomposition [190].

On the other hand, there are some clear instances where Granger Causality will not capture information flow about M correctly, for reasons other than those presented in the counterexample above. Suppose we had four nodes of the computational system A, B, C and D, and A and B had a message M at time $t = 1$, which they transmitted to C and D at time $t = 2$, i.e., $A_1 = B_1 = C_2 = D_2 = M$. In this instance, without knowing the precise edge connectivity between the nodes, it is impossible to tell whether information was transmitted from A to C and B to D, or vice versa, or some combination thereof. Since the M-information flow focuses on measuring edges, such an issue does not arise; however it suffers from the practical difficulty associated with measuring edges. However, Granger causal influence, while providing a measure that depends only on processes at nodes, would simply assign a weight of half to both A and B for each of the receiving nodes C and D in this example. This fundamental difference between what Granger causal influence and M-information flow consider as observed variables makes an objective comparison between the two measures all the more difficult.

Finally, we note that Quinn et al. [31] provide a result showing that conditional directed information can recover the true generative model (i.e., the relationships in the underlying structural causal model). However, to arrive at this result, they rely on an important assumption, namely, that the joint probability distribution is strictly positive everywhere (which is a sufficient condition for satisfying faithfulness in a structural causal model [35]). This assumption disallows perfect redundancy, and therefore prevents us from considering examples such as Counterexample 2.3 from Chapter 2. Therefore, by conditioning on *all* other variables, directed information can recover the true underlying dependency structure,

although the *magnitude* of said dependence may not correctly reflect information flows, since it excludes redundancy.

6.7 Conclusions and discussions

We demonstrate, by means of a concrete counter-example, that the direction predicted by causal influence metrics such as Granger Causality and Directed Information can be opposite to the true direction of information flow. There are, however, several shortcomings to our analysis, which we list in section 6.1.5. We seek to address many of these shortcomings in future work.

It might appear that we make a circular argument while computing the Granger Causal influences in our counter-example, since we supply the model for the stochastic process and use the system parameters of the model directly to compute the Granger Causality metric. However, as we state in Section 6.3.4, we assume that the regression coefficients of the Autoregressive model can be exactly estimated (even if the AR process is non-stationary), and discuss how this might be achieved with the help of multiple trials.

As a final remark, we emphasize that this work only demonstrates the error in *interpreting* the direction of causal influence as the direction of *information flow*. We do *not* seek to invalidate much of the neuroscientific work that has been done in this direction; we merely caution against making (what might be construed as hopeful) extrapolations from causal influences to information flows.

We do not seek to understate the importance of determining causal influences in the brain; understanding causal influence itself may have a great deal of benefit. For instance, we might seek to understand the spread of activity in the brain during an epileptic seizure—in such applications, we are not concerned with how (or what) information is being transferred through the neural circuitry; we only seek to determine the source of the activity for the purpose of surgical intervention.

Acknowledgments

The authors would like to thank Nicola Elia for discussions that suggested this possible misunderstanding in use of Granger causality and Directed Information in computational systems. We also thank Rob Kass, Bruno Sinopoli, Momin Malik and Kun Zhang for useful discussions. Finally, we thank the reviewers of a previous conference submission of this work for their insightful remarks and comments. We thank the CMU BrainHUB initiative for support through a Proseed-BrainHUB seed grant. Pulkit Grover was supported through a CMU startup grant, and Praveen Venkatesh through a CMU CIT Dean's fellowship and a CMU Presidential Fellowship.

6.A Directed Information in the forward direction, noiseless feedback

The Directed Information in the forward direction is computed as:

$$\begin{aligned} I(X^n \to \widehat{\Theta}^n) &= \sum_{i=1}^{n} I(\widehat{\Theta}_i; X^i | \widehat{\Theta}^{i-1}) \\ &= \sum_{i=1}^{n} h(\widehat{\Theta}_i | \widehat{\Theta}^{i-1}) - h(\widehat{\Theta}_i | \widehat{\Theta}^{i-1}, X^i) \end{aligned} \tag{6.32}$$

Taking the first term in (6.32),

$$\begin{aligned} h(\widehat{\Theta}_i | \widehat{\Theta}^{i-1}) &= h\left(\widehat{\Theta}_{i-1} + \frac{X_i + N_i}{i} \middle| \widehat{\Theta}^{i-1} \right) \\ &= h(\Theta - \widehat{\Theta}_{i-1} + N_i | \widehat{\Theta}^{i-1}) - \log(i) \\ &= h(\Theta + N_i | \widehat{\Theta}^{i-1}) - \log(i) \\ &= h(\Theta + N_i | \widehat{\Theta}_{i-1}) - \log(i) \end{aligned}$$

where we have dropped the conditioning on all except $\widehat{\Theta}_{i-1}$ in the last step. Define $U = \Theta + N_i$ and $V = \widehat{\Theta}_{i-1}$. Since all variables are Gaussian, it suffices to find the variance of the conditional distribution $U|V$.

$$\begin{aligned} \mathbb{E}[U] = 0, \mathbb{E}[V] = 0, \mathrm{Var}[U] = \sigma_\theta^2 + \sigma_N^2, \mathrm{Var}[V] &= \sigma_\theta^2 + \frac{\sigma_N^2}{i-1} \\ \mathrm{Cov}[U, V] &= \mathbb{E}[UV] - \mathbb{E}[U]\mathbb{E}[V] \\ &= \sigma_\theta^2 \\ \rho^2 &= \frac{\sigma_\theta^4}{(\sigma_\theta^2 + \sigma_N^2)(\sigma_\theta^2 + \frac{\sigma_N^2}{(i-1)})} \end{aligned}$$

$$U|V = v \ \sim \ \mathcal{N}\left(\sqrt{\frac{\sigma_\theta^2 + \sigma_N^2}{\sigma_\theta^2 + \frac{\sigma_N^2}{i-1}}} \rho v, (1-\rho^2)(\sigma_\theta^2 + \sigma_N^2) \right)$$

Hence, the entropy of the conditional distribution is

$$h(U|V=v) = \frac{1}{2} \log(2\pi e (1-\rho^2)(\sigma_\theta^2 + \sigma_N^2)) \overset{(a)}{=} h(U|V)$$

where (a) follows because the conditional entropy is independent of v. Thus,

$$h(\widehat{\Theta}_i | \widehat{\Theta}^{i-1}) = \frac{1}{2} \log \left(2\pi e \frac{\sigma_N^2 (i\sigma_\theta^2 + \sigma_N^2)}{((i-1)\sigma_\theta^2 + \sigma_N^2)} \right) - \log(i) \tag{6.33}$$

The next term in equation (6.32) is

$$\begin{aligned} h(\widehat{\Theta}_i|\widehat{\Theta}^{i-1}, X^i) &= h\left(\frac{N_i}{i}\middle|\widehat{\Theta}^{i-1}, X^i\right) \\ &= h(N_i) - \log(i) \\ &= \frac{1}{2}\log(2\pi e\sigma_N^2) - \log(i) \end{aligned} \tag{6.34}$$

Putting equations (6.33) and (6.34) together, we can compute the forward Directed Information:

$$\begin{aligned} I(\widehat{\Theta}_i; X^i|\widehat{\Theta}^{i-1}) &= h(\widehat{\Theta}_i|\widehat{\Theta}^{i-1}) - h(\widehat{\Theta}_i|\widehat{\Theta}^{i-1}, X^i) \\ &= \frac{1}{2}\log\left(\frac{i\sigma_\theta^2 + \sigma_N^2}{(i-1)\sigma_\theta^2 + \sigma_N^2}\right) \\ I(X^n \to \widehat{\Theta}^n) &= \sum_{i=1}^{n} I(\widehat{\Theta}_i; X^i|\widehat{\Theta}^{i-1}) \\ &\overset{(a)}{=} \frac{1}{2}\log\left(\frac{n\sigma_\theta^2 + \sigma_N^2}{\sigma_N^2}\right) \\ &= \frac{1}{2}\log\left(1 + \frac{n\sigma_\theta^2}{\sigma_N^2}\right) \end{aligned}$$

where (a) follows through by expanding out the product inside the logarithms and canceling terms. Clearly, this value is finite.

6.B Directed Information in the forward direction, with noisy feedback

$$\begin{aligned} I(X^n \to \widehat{\Theta}^n) &= \sum_{i=1}^{n} I(\widehat{\Theta}_i; X^i|\widehat{\Theta}^{i-1}) \\ &= \sum_{i=1}^{n} h(\widehat{\Theta}_i|\widehat{\Theta}^{i-1}) - h(\widehat{\Theta}_i|\widehat{\Theta}^{i-1}, X^i) \end{aligned}$$

Taking the first of the two terms in the above expression,

$$\begin{aligned} h(\widehat{\Theta}_i|\widehat{\Theta}^{i-1}) &\overset{(a)}{=} h(\widehat{\Theta}_{i-1}\frac{i-1}{i} + \frac{\Theta}{i} - \frac{R_{i-1}}{i} + \frac{N_i}{i}|\widehat{\Theta}^{i-1}) \\ &= h(\Theta - R_{i-1} + N_i|\widehat{\Theta}^{i-1}) - \log(i) \end{aligned}$$

where for (a) we have used equation (6.8). The Markov property no longer holds in this case, but we proceed in the same manner. We define $U = \Theta - R_{i-1} + N_i$ and $\underline{V} = \widehat{\Theta}^{i-1}$. Recalling

equation (6.9), for $j \in \{1, \dots i-1\}$, $p \in \{1, \dots i-1\}$ and $q \in \{1, \dots i-1\}$, we have

$$\begin{aligned}
\mathbb{E}[U] &= 0,\ \mathbb{E}[\widehat{\Theta}_j] = 0, \\
\mathbb{E}[U^2] &= \sigma_\theta^2 + \sigma_R^2 + \sigma_N^2,\ \mathbb{E}[U\widehat{\Theta}_j] = \sigma_\theta^2 \\
\mathbb{E}[\widehat{\Theta}_p \widehat{\Theta}_q] &= \mathbb{E}\Bigg[\left(\Theta + \frac{1}{p}\sum_{k=1}^{p} N_k - \frac{1}{p}\sum_{k=1}^{p-1} R_k\right) \\
&\qquad \left(\Theta + \frac{1}{q}\sum_{k=1}^{q} N_k - \frac{1}{q}\sum_{k=1}^{q-1} R_k\right)\Bigg] \\
&= \sigma_\theta^2 + \frac{\min\{p,q\}}{pq}\sigma_N^2 + \frac{\min\{p-1, q-1\}}{pq}\sigma_R^2 \\
\mathrm{Var}[U|\widehat{\Theta}^{i-1}] &= \mathbb{E}[U^2] - \mathbb{E}[U\underline{V}]\mathbb{E}[\underline{V}\underline{V}^T]^{-1}\mathbb{E}[\underline{V}U] \\
h(U|\widehat{\Theta}^{i-1}) &= \frac{1}{2}\log(2\pi e \mathrm{Var}[U|\widehat{\Theta}^{i-1}])
\end{aligned} \tag{6.35}$$

We can not derive a simple closed form for this expression, but we have computed it numerically for the plots in Section 6.4.1. The second term in equation (6.15) is

$$\begin{aligned}
h(\widehat{\Theta}_i|\widehat{\Theta}^{i-1}, X^i) &= h\left(\widehat{\Theta}_{i-1} + \frac{X_i + N_i}{i}\middle|\widehat{\Theta}^{i-1}, X^i\right) \\
&= h(N_i|\widehat{\Theta}^{i-1}, X^i) - \log(i) \\
&= \frac{1}{2}\log(2\pi e\sigma_N^2) - \log(i)
\end{aligned} \tag{6.36}$$

From equations (6.35) and (6.36), we compute the forward-directed information as depicted in Section 6.4.1, for different values of σ_θ^2.

6.C Directed Information in the reverse direction, with noisy feedback

First, we derive an expression for X_i, which we will use later.

$$\begin{aligned}
X_i &= \Theta - \widehat{\Theta}_{i-1} - R_{i-1} \\
&= \Theta - \left(\Theta + \frac{1}{i-1}\sum_{k=1}^{i-1} N_k - \frac{1}{i-1}\sum_{k=1}^{i-2} R_k\right) - R_{i-1} \\
&= \frac{1}{i-1}\sum_{k=1}^{i-2} R_k - \frac{1}{i-1}\sum_{k=1}^{i-1} N_k - R_i
\end{aligned} \tag{6.37}$$

$$\begin{aligned}
I(0 * \widehat{\Theta}^{n-1} \to X^n) &= \sum_{i=0}^{n-1} I(X_{i+1}; \widehat{\Theta}^i | X^i) \\
&= \sum_{i=0}^{n-1} h(X_{i+1}|X^i) - h(X_{i+1}|X^i, \widehat{\Theta}^i)
\end{aligned} \tag{6.38}$$

Taking the first term inside the summation,

$$\begin{aligned} h(X_{i+1}|X^i) &= h(\Theta - \widehat{\Theta}_i - R_i|X^i) \\ &\overset{(a)}{=} h\bigg(\Big(- \widehat{\Theta}_{i-1} - \frac{X_i + N_i}{i}\Big) - R_i\bigg|X^i\bigg) \\ &\overset{(b)}{=} h\bigg(\Big(X_i - \Theta + R_{i-1} - \frac{N_i}{i}\Big) - R_i\bigg|X^i\bigg) \\ &\overset{(c)}{=} h\bigg(R_{i-1} - \frac{N_i}{i} - R_i\bigg|X^i\bigg) \end{aligned}$$

where in (a) above, we have dropped $\Theta = X_1$, in (b) we have re-expressed $\widehat{\Theta}_{i-1}$ in terms of X_i, Θ and R_{i-1}, and in (c) we have dropped X_i and Θ again. As before, define $U = R_{i-1} - \frac{N_i}{i} - R_i$, so that

$$\mathbb{E}[U] = 0, \;\; \mathbb{E}[X_j] = 0, \;\; \mathbb{E}[U^2] = 2\sigma_R^2 + \frac{\sigma_N^2}{i^2}$$

For $i \geq 3$ and $j \in \{3, \dots i\}$, we can use equation (6.37) to see that

$$\begin{aligned} \mathbb{E}[UX_j] &= \mathbb{E}\bigg[\Big(R_{i-1} - \frac{N_i}{i} - R_i\Big)\Big(\frac{1}{j-1}\sum_{k=1}^{j-2} R_k - \frac{1}{j-1}\sum_{k=1}^{j-1} N_k - R_{j-1}\Big)\bigg] \\ &= -\mathbb{E}[R_{i-1}R_{j-1}] = -\sigma_R^2\delta_{ij} \\ \mathbb{E}[UX_1] &= \mathbb{E}\bigg[\Big(R_{i-1} - \frac{N_i}{i} - R_i\Big)\Theta\bigg] = 0 \\ \mathbb{E}[UX_2] &= \mathbb{E}\bigg[\Big(R_{i-1} - \frac{N_i}{i} - R_i\Big)(\Theta - (\Theta + N_i) - R_1)\bigg] = 0 \\ \mathbb{E}[X_pX_q] &= \mathbb{E}\bigg[\Big(\frac{1}{p-1}\sum_{k=1}^{p-2} R_k - \frac{1}{p-1}\sum_{k=1}^{p-1} N_k - R_{p-1}\Big) \\ &\qquad \Big(\frac{1}{q-1}\sum_{k=1}^{q-2} R_k - \frac{1}{q-1}\sum_{k=1}^{q-1} N_k - R_{q-1}\Big)\bigg] \\ &= \frac{\min\{p-2, q-2\}}{(p-1)(q-1)}\sigma_R^2 + \frac{\min\{p-1, q-1\}}{(p-1)(q-1)}\sigma_N^2 \\ &\qquad + \sigma_R^2\delta_{pq} + \frac{1}{p-1}\sigma_R^2\mathbb{I}_{p>q} + \frac{1}{q-1}\sigma_R^2\mathbb{I}_{q>p} \\ \mathbb{E}[X_1X_1] &= \mathbb{E}[\Theta^2] = \sigma_\theta^2 \\ \mathbb{E}[X_1X_j] &= \mathbb{E}\bigg[\Theta\Big(\frac{1}{j-1}\sum_{k=1}^{j-2} R_k - \frac{1}{j-1}\sum_{k=1}^{j-1} N_k - R_{j-1}\Big)\bigg] = 0 \\ \mathrm{Var}[U|X^i] &= \mathbb{E}[U^2] - \mathbb{E}[UX^i]\mathbb{E}[X^iX^{iT}]^{-1}\mathbb{E}[X^iU] \\ h(U|X^i) &= \frac{1}{2}\log(2\pi e\mathrm{Var}[U|X^i]) \end{aligned} \tag{6.39}$$

The above argument can be extended to $i = 2$ by letting the final index of the summation term be smaller than the starting index, implying that the whole summation term is simply

dropped. The special cases of $i = 0$ and $i = 1$ are handled at the end. The second term from equation (6.16) becomes

$$\begin{aligned} h(X_{i+1}|X^i, \widehat{\Theta}^i) &= h(\Theta - \widehat{\Theta}_i - R_i|X^i, \widehat{\Theta}^i) \\ &= \frac{1}{2}\log(2\pi e\sigma_R^2) \end{aligned} \tag{6.40}$$

because $X_1 = \Theta$ and because R_i is independent of all the X^i and $\widehat{\Theta}^i$. For the special cases of $i = 0$ and $i = 1$, we solve for the value of mutual information explicitly:

$$\begin{aligned} i = 0: \quad & I(X_1; \widehat{\Theta}^0|X^0) && = h(X_1) - h(X_1) = 0 \\ i = 1: \quad & I(X_2; \widehat{\Theta}^1|X^1) && = h(X_2|X_1) - h(X_2|X_1, \widehat{\Theta}_1) \\ & h(X_2|X_1) && = h(\Theta - \widehat{\Theta}_1 - R_1|X^1) \\ & && = h(-\Theta - N_1 - R_1|X_1) \\ & && = h(-N_1 - R_1) \\ & && = \frac{1}{2}\log(2\pi e(\sigma_N^2 + \sigma_R^2)) \\ & h(X_2|X_1, \widehat{\Theta}_1) && = h(-R_1) = \frac{1}{2}\log(2\pi e\sigma_R^2) \\ & I(X_2; \widehat{\Theta}^1|X^1) && = \frac{1}{2}\log\left(2\pi e\frac{\sigma_N^2 + \sigma_R^2}{\sigma_R^2}\right) \end{aligned}$$

From equations (6.39) and (6.40), along with the two special cases above, we can now compute the reverse-directed information plotted in Section 6.4.1.

6.D Derivation of the Markov Chain Failure in Section 6.5

We wish to show that in the canonical example from Section 6.5, M—$[\widehat{M}_1,\ \widehat{M}_2]$—$(M - \widehat{M}_2)$ is not a valid Markov chain. Recall that $Z_1, Z_2, Z_3 \sim$ i.i.d. $\mathcal{N}(0, \sigma^2)$ and $M \sim \mathcal{N}(0, 1)$. Let $h(\cdot)$ denote differential entropy. Then,

$$I(M; M - \widehat{M}_2, \widehat{M}_2) \geq I(M; \widehat{M}_3) = h(\widehat{M}_3) - h(\widehat{M}_3 \,|\, M) \tag{6.41}$$

$$= \frac{1}{2}\log\left(2\pi e\left(1 + \frac{\sigma^2}{3}\right)\right) - \frac{1}{2}\log\left(2\pi e\left(\frac{\sigma^2}{3}\right)\right) \tag{6.42}$$

$$= \frac{1}{2}\log\left(1 + \frac{3}{\sigma^2}\right). \tag{6.43}$$

Here, we started with the Data Processing Inequality, and then used the fact that if $Y \sim \mathcal{N}(0, \sigma^2)$ is a zero-mean scalar Gaussian random variable with variance σ^2, then its differential entropy is given by [97, Thm. 8.4.1]

$$h(Y) = \frac{1}{2}\log(2\pi e\sigma^2) \text{ nats.} \tag{6.44}$$

Next, note that since $\widehat{M}_1 = M + Z_1$ and $\widehat{M}_2 = M + \frac{1}{2}(Z_1 + Z_2)$, $\widehat{M}_1$ has no extra information about M, given $\widehat{M}_2$. This is obvious when we think of $\widehat{M}_1$ as being $\widehat{M}_1 = \widehat{M}_2 + Z'$,

where $Z' = \frac{1}{2}(Z_1 - Z_2)$, and it can be shown that $Z' \perp\!\!\!\perp \widehat{M}_2$:

$$\mathbb{E}[\widehat{M}_2 Z'] = \mathbb{E}\Big[\Big(M + \frac{1}{2}(Z_1 + Z_2)\Big)Z'\Big] \tag{6.45}$$

$$= \mathbb{E}[MZ'] + \frac{1}{4}\mathbb{E}[(Z_1 + Z_2)(Z_1 - Z_2)] \tag{6.46}$$

$$= 0 + \frac{1}{4}\mathbb{E}[Z_1^2 - Z_2^2] \tag{6.47}$$

$$= \frac{1}{4}(\sigma^2 - \sigma^2) = 0. \tag{6.48}$$

Since all variables involved are zero-mean Gaussians, this naturally implies that $\widehat{M}_2 \perp\!\!\!\perp Z'$. Thus, from our previous argument, $\widehat{M}_1$ has no extra information about M when given $\widehat{M}_2$, or in other words, M—$\widehat{M}_2$—$\widehat{M}_1$ is a valid Markov chain. Therefore,

$$I(M; \widehat{M}_1, \widehat{M}_2) = I(M; \widehat{M}_2) + I(M; \widehat{M}_1 \,|\, \widehat{M}_2) \tag{6.49}$$

$$= I(M; \widehat{M}_2) + 0 \tag{6.50}$$

$$= \frac{1}{2}\log\Big(1 + \frac{2}{\sigma^2}\Big), \tag{6.51}$$

derived in the same way as (6.43). From (6.43) and (6.51), we can conclude that $I(M; \widehat{M}_3) > I(M; \widehat{M}_2)$, and therefore

$$I(M; M - \widehat{M}_2, \widehat{M}_2) > I(M; \widehat{M}_1, \widehat{M}_2) \tag{6.52}$$

$$I(M; M - \widehat{M}_2, \widehat{M}_2, \widehat{M}_1) > I(M; \widehat{M}_1, \widehat{M}_2) \tag{6.53}$$

$$I(M; M - \widehat{M}_2, \widehat{M}_2, \widehat{M}_1) - I(M; \widehat{M}_1, \widehat{M}_2) > 0 \tag{6.54}$$

$$I(M; M - \widehat{M}_2 \,|\, \widehat{M}_1, \widehat{M}_2) > 0. \tag{6.55}$$

Thus, the stated Markov chain, M—$[\widehat{M}_1,\ \widehat{M}_2]$—$(M - \widehat{M}_2)$, cannot hold.

7 Discussion

So what I told you was true, from a certain point of view.
— Obi-Wan Kenobi

In this chapter, we discuss a few of the key assumptions and limitations of the M-information flow framework proposed in Chapter 2. We also consider some means by which these assumptions may be satisfied and how limitations may be overcome in future.

7.1 Key Assumptions of the M-Information Flow Framework

7.1.1 Observing edges vs. nodes

The observation model stated in Section 2.5.1 makes a crucial assumption, namely, that transmissions on each *edge* can be observed. In neuroscientific experiments, however, we often record activity from single neurons (as in the case of electrophysiological recordings), or aggregate activity from groups of neurons (as with Local Field Potentials measured in Electrocorticography and Electroencephalography). These neurons, or groups of neurons, are considered to be nodes communicating to one another in a network. It may not be known which nodes are connected to which other nodes, let alone the recipient of each transmission at every time instant. This is a marked departure from our assumption that transmissions on edges can be observed. To some extent, it is possible to incorporate a "node-centric" model within our computational system by assuming that all nodes broadcast their transmissions. However, that still leaves unanswered the question of which nodes actually "hear" another's transmissions. A possible resolution to that question might arise from an understanding of *receiver response*. That is, we consider a revised model in which an edge exists if a receiving neuron *uses* the information transmitted by some neuron at the previous time instant. This issue is beyond the scope of the current work, and will be addressed in subsequent studies.

We also note that, although tools based on Granger Causality implicitly assume that nodes are measured and not edges, they do not resolve the issue of which node is "talking" to which other node. For example, if two different nodes A_1 and B_1 communicate the same information to a third node, C_2, any regression based analysis will assign a weight of one-half to each of A_1 and B_1. However, the true function, f_{C_2}, may be using only the information

coming from A_1, or only the information coming from B_1, or using the two in some other unequal proportion. Such cases may only be identifiable through an interventional approach.

Conversely, our work may suggest to neuroscientists that inferences about information flow are more reliably obtained if one can measure transmissions on edges in the graph, rather than transmissions of nodes. This may call for newer imaging modalities, or new uses of existing modalities, such as treating axons as targets for invasive recordings, perhaps at nodes of Ranvier. Some work along these lines has already appeared: Patolsky et al. [191] develop new techniques to measure signals along the length of an axon. Such ideas may need to be revisited in greater detail, given the importance of measuring edges.

7.1.2 Observing memories

Another important assumption in the observation model is that *memories* of nodes are observed as transmissions on self-edges. If these transmissions are implemented in the form of some internal state at each node, then they might be difficult to observe in practice.[1]

It remains to be fully understood whether one can compensate for not observing memories in some manner, e.g., by assuming that the memory of a node is the full history of its transmissions and receptions. While this means that intrinsically generated random variables that are *not* propagated to other nodes will never be observed, it could be argued that such variables could have no impact on the system (save for acting as "computational noise"). So perhaps it suffices to observe only transmissions between *different* nodes (and not self-edges). Further work is required to understand what ramifications such an assumption has on identifying information flows and information paths.

Alternatively, perhaps if one wishes to observe memories, it is important to measure not only spikes, but also membrane voltages (e.g. using voltage-sensitive dyes [136] or, less directly, through measurements of changes in neurotransmitter concentrations outside a cell [192]).

7.1.3 Discretization of time

Yet another implicit assumption in our computational system model is that transmissions occur at discrete points in time. This assumption is justified for synchronous digital circuits used commonly today, or if the computational system of interest is a trained artificial neural network, for instance. However, this is not a perfect model of the brain, because neural spiking (among other processes), does not occur only at multiples of some fundamental unit of time. The same also holds true of dendritic and axonal propagation delays, for instance.

This issue might be partially mitigated by assuming that neural computation happens at a certain time scale, and by using a sufficiently high sampling rate so that Nyquist-rate–type arguments apply. However, Nyquist-rate sampling may not be possible in certain modalities that are inherently slow (e.g. Calcium imaging and functional Magnetic Resonance Imaging), so it would be interesting to understand what inferences we are no longer capable of making. Alternatively, if the sampling rate is too high, it may be useful to look for M-information

[1] If every node represents a *group* of neurons, however it may just be that their internal state is represented in the form of *communication between these neurons*. In that case, perhaps observing their internal state is just a matter of having more spatially refined measurements.

flow within *time windows*, which could help increase the sample size for detection. The exact implications of using such preprocessing methods will also need to be studied in greater detail, and forms another avenue for future work.

7.1.4 Message enters at $t = 0$

Another assumption in our framework is that the message enters the system at, and only at, time $t = 0$. This is essential, given the way we have defined input nodes: nodes at time $t = 0$, whose outputs depend on the message (and which have no other shared source of randomness). However, this assumption does not allow for a dynamically evolving stimulus, which is also common in neuroscientific experiments.

Suppose we allow the message to enter the system at a later time instant, say at some node U_t, for $t > 0$, i.e., U_t may compute a function not just of its inputs, but also of M. Then, if we want the information path theorem to continue to hold, we must also add U_t to the set of input nodes.[2] Thus, if we see dependence at some other node $V_{t'}$, at a later time instant $t' > t$, the information paths leading to $V_{t'}$ may arise from the original input nodes *or* from U_t, or both. As we might intuitively expect, the more time points we allow the message to enter *at*, the more such information paths we will likely see, making the results of our analysis harder to interpret.

On a related note, recall that the assumption that M enters only at $t = 0$ comes from our decision to focus on event related experimental paradigms [65] (refer Section 2.1.2). However, although neural responses to event related stimuli are often time-locked as well, they have considerable variability: i.e., a neuron may respond at random times in each trial. Apart from the inherent stochasticity in neural firing, this could happen for any number of reasons, including the animal's state of arousal, its attention levels in each trial, etc. In our framework, the information flow will be smeared out over the entire time interval of possible response. A modified computational model may be needed to address such an issue.

7.1.5 Experimental design and the message

An important aspect of our work is that it explicitly incorporates the message, which in neuroscientific experiments is often some information contained in the stimulus. This aids the neuroscientist in designing experiments, for example, in understanding what stimuli will help them make a certain inference about information flow. In particular, one needs to use at least two different stimuli in order to obtain any determination about information flow. While this is implicitly understood in neuroscience, as evidenced by comparisons with baselines, or by the use of permutation tests to scramble stimulus-trial correlations for a null model, our framework provides a more direct method for identifying and interpreting stimulus-related information flow. This could be particularly useful when tracking the flows of multiple messages individually (see Section 2.5.6).

[2] We should also expect that any Local Markovity conditions at time t (see Proposition 2.6) that involve the node U_t will no longer hold.

7.2 Limitations of the Model

It is also worth discussing some aspects of the brain, and of information flow (broadly understood), which our model does not capture. Here, we may require different or more specialized models that are tailored to the specifics of the application in order to infer information flows. However, we believe that our model can still serve as a useful baseline for the design of such specialized models.

7.2.1 Non-independent trials, learning and changes to the network

Firstly, trials are assumed to be independent and identically distributed, with similar start times. However, neuroscientific experiments often have trials that are presented serially, over which time the brain's attention levels can wax and wane. The animal or human participant may also learn how to perform the task better over time, leading to a significant difference in trials in a later block, compared to an earlier block. The M-information flow framework does not explicitly account for such possibilities; if learning, adaptation or attention are the very subject of study in a particular experiment, then the message may have to be assigned very carefully when applying such a framework.

The computational model also assumes to some degree a static underlying network, where the structure of the circuit does not change over time. Such changes may arise in experiments analyzing learning over longer time-scales, or in experiments where the brain suffers acute injury [9] or recovery from a disease [25, 45]. A recent work of the author considers how one might *detect* whether a network's connectivity structure has changed [193]. While this work does not answer the overarching question of tracking information flow, it could form the basis for determining whether a different model is necessary in a specific context.

7.2.2 Accounting for axonal delays

Another aspect of neural computation that our computational model does not account for is axonal delay. In particular, we assume that computations are synchronous and that all edges relay transmissions with the same delays to their receiving nodes. However, axons may have different lengths and conduction speeds, which could create variable delays in transmitting information between different pairs of nodes. In practice, if the time-scale at which computations occur is much slower than the axonal delays (e.g., because said computation requires aggregation of spikes over time), then this may not be relevant to inferring information flows.

There are, however, instances where axonal delays play essential roles in the computation itself: an excellent example of this is the auditory localization circuit [194]. In such cases, to fit the computational model, we would need to know where the message is at time scales much faster than axonal conduction. In other words, each axon would have to be divided into multiple nodes, and we would need to track information flow along its length. While there have been some efforts at designing neurotechnologies to achieve such measurements [191], such experiments may need a more carefully designed (or even specialized) model for inferring information flow.

7.2.3 Tracking information flows not tied to a message

However, our definition of information flow does not cater to instances where the information is *not specific to a message*. This could refer to cases such as the spread of electrical activity during a seizure. Here, the message is not a well-defined entity, nevertheless there is a flow of "information" in the abstract, away from the seizure focus. Understanding the paths along which this abstract information flows could be useful for locating the origin of spread.

While our computational model may still apply to such a setting, it may be altogether too detailed for capturing flows in an environment where we are not particularly concerned about computation. Models of spreading phenomena from network science [59, 195] may be more appropriate for analyzing such flows. It is also possible that tools such as Granger Causality can perform the role of measuring such flows adequately, since they are more similar to causal influences, however, assessing this rigorously is left to future work.

Bibliography

[1] I. H. Stevenson and K. P. Kording, "How advances in neural recording affect data analysis," *Nature neuroscience*, vol. 14, no. 2, pp. 139–142, 2011.

[2] A. E. Urai, B. Doiron, A. M. Leifer, and A. K. Churchland, "Large-scale neural recordings call for new insights to link brain and behavior," *arXiv preprint arXiv:2103.14662*, 2021.

[3] P. Gao and S. Ganguli, "On simplicity and complexity in the brave new world of large-scale neuroscience," *Current opinion in neurobiology*, vol. 32, pp. 148–155, 2015.

[4] T. P. Lillicrap and K. P. Kording, "What does it mean to understand a neural network?" *arXiv preprint arXiv:1907.06374*, 2019.

[5] C. Molnar, *Interpretable Machine Learning*, 2019, https://christophm.github.io/interpretable-ml-book/.

[6] D. Marr and T. Poggio, "From understanding computation to understanding neural circuitry," *AI Memos*, 1976. [Online]. Available: http://hdl.handle.net/1721.1/5782

[7] P. S. Churchland and T. J. Sejnowski, "Perspectives on cognitive neuroscience," *Science*, vol. 242, no. 4879, pp. 741–745, 1988.

[8] J. Almeida, A. R. Fintzi, and B. Z. Mahon, "Tool manipulation knowledge is retrieved by way of the ventral visual object processing pathway," *Cortex*, vol. 49, no. 9, pp. 2334–2344, 2013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0010945213001329

[9] Y. K. Hong, C. O. Lacefield, C. C. Rodgers, and R. M. Bruno, "Sensation, movement and learning in the absence of barrel cortex," *Nature*, vol. 561, no. 7724, pp. 542–546, 2018.

[10] A. Brovelli, M. Ding, A. Ledberg, Y. Chen, R. Nakamura, and S. L. Bressler, "Beta oscillations in a large-scale sensorimotor cortical network: Directional influences revealed by Granger causality," *Proceedings of the National Academy of Sciences*, vol. 101, no. 26, pp. 9849–9854, 2004. [Online]. Available: http://www.pnas.org/content/101/26/9849

[11] M. Bar, K. S. Kassam, A. S. Ghuman, J. Boshyan, A. M. Schmid, A. M. Dale, M. S. Hämäläinen, K. Marinkovic, D. L. Schacter, B. R. Rosen, and E. Halgren, "Top-down facilitation of visual recognition," *Proceedings of the National Academy of Sciences*, vol. 103, no. 2, pp. 449–454, 2006. [Online]. Available: https://www.pnas.org/content/103/2/449

[12] A. S. Greenberg, T. Verstynen, Y.-C. Chiu, S. Yantis, W. Schneider, and M. Behrmann, "Visuotopic cortical connectivity underlying attention revealed with white-matter tractography," *Journal of Neuroscience*, vol. 32, no. 8, pp. 2773–2782, 2012. [Online]. Available: http://www.jneurosci.org/content/32/8/2773

[13] C. Hammond, H. Bergman, and P. Brown, "Pathological synchronization in Parkinson's disease: networks, models and treatments," *Trends in neurosciences*, vol. 30, no. 7, pp. 357–364, 2007.

[14] E. Lalo, S. Thobois, A. Sharott, G. Polo, P. Mertens, A. Pogosyan, and P. Brown, "Patterns of bidirectional communication between cortex and basal ganglia during movement in patients with Parkinson disease," *Journal of Neuroscience*, vol. 28, no. 12, pp. 3008–3016, 2008.

[15] J. a. D. Semedo, A. Zandvakili, C. K. Machens, M. Y. Byron, and A. Kohn, "Cortical areas interact through a communication subspace," *Neuron*, vol. 102, no. 1, pp. 249–259, 2019.

[16] Y. Smith, M. Bevan, E. Shink, and J. Bolam, "Microcircuitry of the direct and indirect pathways of the basal ganglia," *Neuroscience*, vol. 86, no. 2, pp. 353–387, 1998.

[17] A. A. Grace, "Gating of information flow within the limbic system and the pathophysiology of schizophrenia," *Brain Research Reviews*, vol. 31, no. 2-3, pp. 330–341, 2000.

[18] D. C. Hammond, "Neurofeedback treatment of depression and anxiety," *Journal of Adult Development*, vol. 12, no. 2-3, pp. 131–137, 2005.

[19] ——, "QEEG-guided neurofeedback in the treatment of obsessive compulsive disorder," *Journal of Neurotherapy*, vol. 7, no. 2, pp. 25–52, 2003.

[20] E. H. Kossoff, E. K. Ritzl, J. M. Politsky, A. M. Murro, J. R. Smith, R. B. Duckrow, D. D. Spencer, and G. K. Bergey, "Effect of an external responsive neurostimulator on seizures and electrographic discharges during subdural electrode monitoring," *Epilepsia*, vol. 45, no. 12, pp. 1560–1567, 2004.

[21] J. S. Perlmutter and J. W. Mink, "Deep brain stimulation," *Annual Review of Neuroscience*, vol. 29, no. 1, pp. 229–257, 2006, pMID: 16776585. [Online]. Available: https://doi.org/10.1146/annurev.neuro.29.051605.112824

[22] H. S. Mayberg, A. M. Lozano, V. Voon, H. E. McNeely, D. Seminowicz, C. Hamani, J. M. Schwalb, and S. H. Kennedy, "Deep brain stimulation for treatment-resistant depression," *Neuron*, vol. 45, no. 5, pp. 651–660, 2005.

[23] N. D. Volkow and F. S. Collins, "The role of science in addressing the opioid crisis," *New England Journal of Medicine*, vol. 377, no. 4, pp. 391–394, 2017.

[24] J. Cao and P. Grover, "Stimulus: Noninvasive dynamic patterns of neurostimulation using spatio-temporal interference," *IEEE Transactions on Biomedical Engineering*, vol. 67, no. 3, pp. 726–737, 2019.

[25] P. Venkatesh, D. Sneider, M. Danish, N. D. Sisterson, N. Zaher, A. Urban, P. Grover, R. M. Richardson, and V. Kokkinos, "Quantifying a frequency modulation response biomarker in responsive neurostimulation," *Journal of Neural Engineering*, mar 2021. [Online]. Available: https://doi.org/10.1088/1741-2552/abed82

[26] C. W. J. Granger, "Investigating causal relations by econometric models and cross-spectral methods," *Econometrica: Journal of the Econometric Society*, pp. 424–438, 1969.

[27] S. L. Bressler and A. K. Seth, "Wiener–Granger causality: A well established methodology," *NeuroImage*, vol. 58, no. 2, pp. 323–329, 2011. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1053811910002272

[28] T. Schreiber, "Measuring information transfer," *Physical Review Letters*, vol. 85, pp. 461–464, Jul 2000. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevLett.85.461

[29] J. Massey, "Causality, feedback and directed information," in *Proceedings of the International Symposium on Information Theory and its Applications (ISITA)*, 1990, pp. 303–305.

[30] C. J. Quinn, T. P. Coleman, N. Kiyavash, and N. G. Hatsopoulos, "Estimating the directed information to infer causal relationships in ensemble neural spike train recordings," *Journal of Computational Neuroscience*, vol. 30, no. 1, pp. 17–44, Feb 2011. [Online]. Available: https://doi.org/10.1007/s10827-010-0247-2

[31] C. J. Quinn, N. Kiyavash, and T. P. Coleman, "Directed information graphs," *IEEE Transactions on Information Theory*, vol. 61, no. 12, pp. 6887–6909, Dec 2015.

[32] J. Jiao, H. H. Permuter, L. Zhao, Y. Kim, and T. Weissman, "Universal estimation of directed information," *IEEE Transactions on Information Theory*, vol. 59, no. 10, pp. 6220–6242, Oct 2013.

[33] Y. Lazebnik, "Can a biologist fix a radio?—Or, what I learned while studying apoptosis," *Cancer cell*, vol. 2, no. 3, pp. 179–182, 2002.

[34] J. Pearl, *Causality: Models, Reasoning and Inference*. Cambridge University Press, 2009.

[35] J. Peters, D. Janzing, and B. Schölkopf, *Elements of causal inference: Foundations and learning algorithms*. MIT press, 2017.

[36] P. Nunez, , and R. Srinivasan, *Electric Fields of the Brain: The Neurophysics of EEG*. Oxford University Press, 2006.

[37] P. Grover and P. Venkatesh, "An information-theoretic view of EEG sensing," *Proceedings of the IEEE*, vol. 105, no. 2, pp. 367–384, 2016.

[38] P. Venkatesh and P. Grover, "Lower bounds on the minimax risk for the source localization problem," in *2017 IEEE International Symposium on Information Theory (ISIT)*, June 2017, pp. 3080–3084.

[39] A. Xu and T. Coleman, "Minimax lower bounds for circular source localization," in *2020 IEEE International Symposium on Information Theory (ISIT)*. IEEE, 2020, pp. 1242–1247.

[40] A. K. Robinson, P. Venkatesh, M. J. Boring, M. J. Tarr, P. Grover, and M. Behrmann, "Very high density EEG elucidates spatiotemporal aspects of early visual processing," *Scientific Reports*, vol. 7, no. 1, pp. 1–11, 2017.

[41] Z. Ahmed, J. Reddy, K. Deshpande, A. Krishnan, P. Venkatesh, S. Kelly, P. Grover, and M. Chamanzar, "Flexible ultra-resolution subdermal EEG probes," in *2018 IEEE Biomedical Circuits and Systems Conference (BioCAS)*. IEEE, 2018, pp. 1–4.

[42] A. Krishnan, R. Kumar, P. Venkatesh, S. Kelly, and P. Grover, "Low-cost carbon fiber-based conductive silicone sponge EEG electrodes," in *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2018, pp. 1287–1290.

[43] P. Grover, J. A. Weldon, S. K. Kelly, P. Venkatesh, and H. Jeong, "An information theoretic technique for harnessing attenuation of high spatial frequencies to design ultra-high-density EEG," in *2015 53rd Annual Allerton Conference on Communication, Control, and Computing (Allerton)*. IEEE, 2015, pp. 901–908.

[44] A. Chamanzar, S. George, P. Venkatesh, M. Chamanzar, L. Shutter, J. Elmer, and P. Grover, "An algorithm for automated, noninvasive detection of cortical spreading depolarizations based on EEG simulations," *IEEE Transactions on Biomedical Engineering*, vol. 66, no. 4, pp. 1115–1126, 2018.

[45] V. Kokkinos, N. D. Sisterson, T. A. Wozny, and R. M. Richardson, "Association of closed-loop brain stimulation neurophysiological features with seizure control among patients with focal epilepsy," *JAMA neurology*, vol. 76, no. 7, pp. 800–808, 2019.

[46] S. Dutta, P. Venkatesh, P. Mardziel, A. Datta, and P. Grover, "An information-theoretic quantification of discrimination with exempt features," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 04, 2020, pp. 3825–3833.

[47] ——, "Fairness under feature exemptions: Counterfactual and observational measures," *arXiv preprint arXiv:2006.07986*, 2020.

[48] G. Schamberg and P. Venkatesh, "Partial information decomposition via deficiency for multivariate gaussians," *arXiv e-prints*, pp. arXiv–2105, 2021, equal contribution.

[49] P. Venkatesh, S. Dutta, and P. Grover, "Information flow in computational systems," *IEEE Transactions on Information Theory*, vol. 66, no. 9, pp. 5456–5491, September 2020. [Online]. Available: https://doi.org/10.1109/TIT.2020.2987806

[50] R. Ahlswede, N. Cai, S. Y. R. Li, and R. W. Yeung, "Network information flow," *IEEE Transactions on Information Theory*, vol. 46, no. 4, pp. 1204–1216, July 2000.

[51] M. Franceschetti and R. Meester, *Random networks for communication: from statistical physics to information systems*. Cambridge University Press, 2008, vol. 24.

[52] P. Grover, "Actions can speak more clearly than words," Ph.D. dissertation, University of California, Berkeley, 2010.

[53] G. Ranade, "Active systems with uncertain parameters: an information-theoretic perspective," Ph.D. dissertation, University of California, Berkeley, 2014.

[54] G. Smith, "On the foundations of quantitative information flow," in *International Conference on Foundations of Software Science and Computational Structures*. Springer, 2009, pp. 288–302.

[55] A. Datta, S. Sen, and Y. Zick, "Algorithmic transparency via quantitative input influence: Theory and experiments with learning systems," in *IEEE Symposium on Security and Privacy*, 2016, pp. 598–617.

[56] Z. Goldfeld, E. v. d. Berg, K. Greenewald, I. Melnyk, N. Nguyen, B. Kingsbury, and Y. Polyanskiy, "Estimating information flow in deep neural networks," in *Proceedings of the 36th International Conference on Machine Learning*, 2019, pp. 2299–2308.

[57] N. Tishby, F. C. Pereira, and W. Bialek, "The information bottleneck method," *arXiv preprint physics/0004057*, 2000.

[58] N. Tishby and N. Zaslavsky, "Deep learning and the information bottleneck principle," in *2015 IEEE Information Theory Workshop (ITW)*. IEEE, 2015, pp. 1–5.

[59] M. Newman, *Networks*. Oxford University Press, 2018. [Online]. Available: https://books.google.com/books?id=YdZjDwAAQBAJ

[60] Y. LeCun, J. S. Denker, and S. A. Solla, "Optimal brain damage," in *Advances in Neural Information Processing Systems*, 1990, pp. 598–605.

[61] J. Frankle and M. Carbin, "The lottery ticket hypothesis: Finding sparse, trainable neural networks," in *7th International Conference on Learning Representations, ICLR 2019*, 2019. [Online]. Available: https://openreview.net/forum?id=rJl-b3RcF7

[62] A. L. Benabid, "Deep brain stimulation for Parkinson's disease," *Current opinion in neurobiology*, vol. 13, no. 6, pp. 696–706, 2003.

[63] G. Clark, "Cochlear implants," in *Speech processing in the auditory system*. Springer, 2004, pp. 422–462.

[64] J. D. Weiland, W. Liu, and M. S. Humayun, "Retinal prosthesis," *Annu. Rev. Biomed. Eng.*, vol. 7, pp. 361–401, 2005.

[65] J. Samuels and N. D. Zasler, *Event-Related Paradigms*. Springer, 2018, pp. 1346–1347.

[66] C. D. Thompson, "A complexity theory for VLSI," Ph.D. dissertation, Carnegie Mellon University, Pittsburgh, PA, USA, 1980, aAI8100621.

[67] E. Jonas and K. P. Kording, "Could a neuroscientist understand a microprocessor?" *PLoS computational biology*, vol. 13, no. 1, p. e1005268, 2017.

[68] K. J. Friston, "Functional and effective connectivity: a review," *Brain connectivity*, vol. 1, no. 1, pp. 13–36, 2011.

[69] K. Friston, R. Moran, and A. K. Seth, "Analysing connectivity with Granger causality and dynamic causal modelling," *Current opinion in neurobiology*, vol. 23, no. 2, pp. 172–178, 2013.

[70] A. M. Bastos and J.-M. Schoffelen, "A tutorial review of functional connectivity analysis methods and their interpretational pitfalls," *Frontiers in systems neuroscience*, vol. 9, p. 175, 2016.

[71] L. A. Baccalá and K. Sameshima, "Partial directed coherence: a new concept in neural structure determination," *Biological Cybernetics*, vol. 84, no. 6, pp. 463–474, May 2001. [Online]. Available: https://doi.org/10.1007/PL00007990

[72] O. David, I. Guillemain, S. Saillet, S. Reyt, C. Deransart, C. Segebarth, and A. Depaulis, "Identifying neural drivers with functional MRI: an electrophysiological validation," *PLoS biology*, vol. 6, no. 12, p. e315, 2008.

[73] A. Roebroeck, E. Formisano, and R. Goebel, "The identification of interacting networks in the brain using fMRI: model selection, causality and deconvolution," *Neuroimage*, vol. 58, no. 2, pp. 296–302, 2011.

[74] O. David, "fMRI connectivity, meaning and empiricism. comments on: Roebroeck et al. The identification of interacting networks in the brain using fMRI: model selection, causality and deconvolution." *Neuroimage*, vol. 58, no. 2, pp. 306–309, 2011.

[75] P. A. Stokes and P. L. Purdon, "A study of problems encountered in Granger causality analysis from a neuroscience perspective," *Proceedings of the National Academy of Sciences*, vol. 114, no. 34, pp. E7063–E7072, 2017. [Online]. Available: http://www.pnas.org/content/114/34/E7063

[76] L. Barnett, A. B. Barrett, and A. K. Seth, "Solved problems for Granger causality in neuroscience: A response to Stokes and Purdon," *NeuroImage*, vol. 178, pp. 744–748, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1053811918304932

[77] L. Faes, S. Stramaglia, and D. Marinazzo, "On the interpretability and computational reliability of frequency-domain Granger causality," *F1000Research*, Sep 2017.

[78] P. A. Stokes and P. L. Purdon, "In reply to Faes et al. and Barnett et al. regarding "a study of problems encountered in Granger causality analysis from a neuroscience perspective"," *arXiv:1709.10248 [stat.ME]*, 2017.

[79] J. Andersson, "Testing for Granger causality in the presence of measurement errors," *Economics Bulletin*, 2005.

[80] H. Nalatore, M. Ding, and G. Rangarajan, "Mitigating the effects of measurement noise on Granger causality," *Physical Review E*, vol. 75, no. 3, p. 031123, Mar 2007. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevE.75.031123

[81] M. Gong, K. Zhang, B. Schölkopf, D. Tao, and P. Geiger, "Discovering temporal causal relations from subsampled data," in *Proceedings of The 32nd International Conference on Machine Learning*, vol. 37, Jul 2015, pp. 1898–1906. [Online]. Available: http://proceedings.mlr.press/v37/gongb15.html

[82] P. Venkatesh and P. Grover, "Is the direction of greater Granger causal influence the same as the direction of information flow?" in *53rd Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, Sept 2015, pp. 672–679.

[83] C. E. Shannon, "A mathematical theory of communication," *The Bell System Technical Journal*, vol. 27, no. 3, pp. 379–423, July 1948.

[84] A. Sheikhattar, S. Miran, J. Liu, J. B. Fritz, S. A. Shamma, P. O. Kanold, and B. Babadi, "Extracting neuronal functional network dynamics via adaptive Granger causality analysis," *Proceedings of the National Academy of Sciences*, vol. 115, no. 17, pp. E3869–E3878, 2018.

[85] K. J. Friston, L. Harrison, and W. Penny, "Dynamic causal modelling," *Neuroimage*, vol. 19, no. 4, pp. 1273–1302, 2003.

[86] A. V. Oppenheim, J. R. Buck, and R. W. Schafer, *Discrete-time signal processing*, 2nd ed. Upper Saddle River, N.J.: Prentice Hall, 1999.

[87] Y. Polyanskiy and Y. Wu, "Lecture notes on information theory," August 2017. [Online]. Available: http://www.stat.yale.edu/~yw562/teaching/itlectures.pdf

[88] C. E. Shannon, "Communication theory of secrecy systems," *Bell system technical journal*, vol. 28, no. 4, pp. 656–715, 1949.

[89] P. L. Williams and R. D. Beer, "Nonnegative decomposition of multivariate information," *arXiv:1004.2515 [cs.IT]*, 2010.

[90] M. Harder, C. Salge, and D. Polani, "Bivariate measure of redundant information," *Phys. Rev. E*, vol. 87, p. 012130, Jan 2013. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevE.87.012130

[91] N. Bertschinger, J. Rauh, E. Olbrich, J. Jost, and N. Ay, "Quantifying unique information," *Entropy*, vol. 16, no. 4, pp. 2161–2183, 2014. [Online]. Available: http://www.mdpi.com/1099-4300/16/4/2161

[92] J. T. Lizier, N. Bertschinger, J. Jost, and M. Wibral, "Information decomposition of target effects from multi-source interactions: Perspectives on previous, current and future work," *Entropy*, vol. 20, no. 4, p. 307, 2018.

[93] E. Schneidman, W. Bialek, and M. J. Berry, "Synergy, redundancy, and independence in population codes," *Journal of Neuroscience*, vol. 23, no. 37, pp. 11 539–11 553, 2003.

[94] P. E. Latham and S. Nirenberg, "Synergy, redundancy, and independence in population codes, revisited," *Journal of Neuroscience*, vol. 25, no. 21, pp. 5195–5206, 2005.

[95] N. M. Timme and C. Lapish, "A tutorial for information theory in neuroscience," *eNeuro*, vol. 5, no. 3, 2018.

[96] I. Gat and N. Tishby, "Synergy and redundancy among brain cells of behaving monkeys," in *Advances in Neural Information Processing Systems*, 1999, pp. 111–117.

[97] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. John Wiley & Sons, 2012.

[98] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, 3rd ed. The MIT Press, 2009.

[99] W. P. Bergsma, "Testing conditional independence for continuous random variables," *EURANDOM report*, vol. 2004, no. 049, 2004.

[100] K. Zhang, J. Peters, D. Janzing, and B. Schölkopf, "Kernel-based conditional independence test and application in causal discovery," in *Proceedings of the Twenty-Seventh Conference on Uncertainty in Artificial Intelligence*, ser. UAI'11. Arlington, Virginia, United States: AUAI Press, 2011, pp. 804–813. [Online]. Available: http://dl.acm.org/citation.cfm?id=3020548.3020641

[101] T.-M. Huang *et al.*, "Testing conditional independence using maximal nonlinear conditional correlation," *The Annals of Statistics*, vol. 38, no. 4, pp. 2047–2091, 2010.

[102] L. Su and H. White, "A consistent characteristic function-based test for conditional independence," *Journal of Econometrics*, vol. 141, no. 2, pp. 807–834, 2007. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0304407606002375

[103] M. Huang, Y. Sun, and H. White, "A flexible nonparametric test for conditional independence," *Econometric Theory*, vol. 32, no. 6, pp. 1434–1482, 2016.

[104] R. Sen, K. Shanmugam, H. Asnani, A. Rahimzamani, and S. Kannan, "Mimic and classify: A meta-algorithm for conditional independence testing," 2018.

[105] R. D. Shah and J. Peters, "The hardness of conditional independence testing and the generalised covariance measure," *arXiv:1804.07203 [math.ST]*, Apr 2018. [Online]. Available: https://arxiv.org/abs/1804.07203

[106] L. Paninski, "Estimation of entropy and mutual information," *Neural computation*, vol. 15, no. 6, pp. 1191–1253, 2003.

[107] W. Gao, S. Kannan, S. Oh, and P. Viswanath, "Estimating mutual information for discrete-continuous mixtures," in *Advances in Neural Information Processing Systems*, 2017, pp. 5986–5997.

[108] A. Kraskov, H. Stögbauer, and P. Grassberger, "Estimating mutual information," *Physical review E*, vol. 69, no. 6, p. 066138, 2004.

[109] H. Liu, L. Wasserman, and J. D. Lafferty, "Exponential concentration for mutual information estimation with application to forests," in *Advances in Neural Information Processing Systems*, 2012, pp. 2537–2545.

[110] J.-P. Lachaux, E. Rodriguez, J. Martinerie, and F. J. Varela, "Measuring phase synchrony in brain signals," *Human brain mapping*, vol. 8, no. 4, pp. 194–208, 1999.

[111] F. Varela, J.-P. Lachaux, E. Rodriguez, and J. Martinerie, "The brainweb: phase synchronization and large-scale integration," *Nature reviews neuroscience*, vol. 2, no. 4, p. 229, 2001.

[112] M. I. Posner, "Orienting of attention," *Quarterly journal of experimental psychology*, vol. 32, no. 1, pp. 3–25, 1980.

[113] D. S. Bassett and E. Bullmore, "Small-world brain networks," *The Neuroscientist*, vol. 12, no. 6, pp. 512–523, 2006.

[114] S. Achard, R. Salvador, B. Whitcher, J. Suckling, and E. Bullmore, "A resilient, low-frequency, small-world human brain functional network with highly connected association cortical hubs," *Journal of Neuroscience*, vol. 26, no. 1, pp. 63–72, 2006.

[115] S. J. Russo and E. J. Nestler, "The brain reward circuitry in mood disorders," *Nature Reviews Neuroscience*, vol. 14, no. 9, p. 609, 2013.

[116] J. P. Shaffer, "Multiple hypothesis testing," *Annual review of psychology*, vol. 46, no. 1, pp. 561–584, 1995.

[117] B. Duan, A. Ramdas, S. Balakrishnan, and L. Wasserman, "Interactive martingale tests for the global null," *arXiv:1909.07339 [stat.ME]*, 2019.

[118] Y. Benjamini and Y. Hochberg, "Controlling the false discovery rate: A practical and powerful approach to multiple testing," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 57, no. 1, pp. 289–300, 1995. [Online]. Available: http://www.jstor.org/stable/2346101

[119] D. Koller, N. Friedman, and F. Bach, *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

[120] N. Ay and D. Polani, "Information flows in causal networks," *Advances in Complex Systems*, vol. 11, no. 01, pp. 17–41, 2008. [Online]. Available: https://doi.org/10.1142/S0219525908001465

[121] D. Janzing, D. Balduzzi, M. Grosse-Wentrup, and B. Schölkopf, "Quantifying causal influences," *Ann. Statist.*, vol. 41, no. 5, pp. 2324–2358, 10 2013. [Online]. Available: https://doi.org/10.1214/13-AOS1145

[122] P. Venkatesh and P. Grover, "Understanding encoding and redundancy in grid cells using partial information decomposition," in *Computational and Systems Neuroscience (Cosyne)*, 2020. [Online]. Available: https://praveenv253.github.io/publications#Venkatesh2020Understanding

[123] P. Venkatesh, S. Dutta, and P. Grover, "How else can we define information flow in neural circuits?" in *2020 IEEE International Symposium on Information Theory (ISIT)*. IEEE, 2020, pp. 2879–2884.

[124] P. K. Banerjee, E. Olbrich, J. Jost, and J. Rauh, "Unique informations and deficiencies," in *2018 56th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*. IEEE, 2018, pp. 32–38.

[125] P. K. Banerjee, J. Rauh, and G. Montúfar, "Computing the unique information," in *2018 IEEE International Symposium on Information Theory (ISIT)*. IEEE, 2018, pp. 141–145.

[126] G. Pica, E. Piasini, H. Safaai, C. Runyan, C. Harvey, M. Diamond, C. Kayser, T. Fellin, and S. Panzeri, "Quantifying how much sensory information in a neural code is relevant for behavior," in *Advances in Neural Information Processing Systems*, 2017, pp. 3686–3696.

[127] S. Sreenivasan and I. Fiete, "Grid cells generate an analog error-correcting code for singularly precise neural computation," *Nature neuroscience*, vol. 14, no. 10, p. 1330, 2011.

[128] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. John Wiley & Sons, 2012.

[129] V. Griffith and C. Koch, *Quantifying Synergistic Mutual Information*. Springer Berlin Heidelberg, 2014, pp. 159–190.

[130] B. Ermentrout and N. Kopell, "Parabolic bursting in an excitable system coupled with a slow oscillation," *SIAM*, vol. 46, no. 2, pp. 233–253, 1986.

[131] A. Gidon, T. A. Zolnik, P. Fidzinski, F. Bolduan, A. Papoutsi, P. Poirazi, M. Holtkamp, I. Vida, and M. E. Larkum, "Dendritic action potentials and computation in human layer 2/3 cortical neurons," *Science*, vol. 367, no. 6473, pp. 83–87, 2020. [Online]. Available: https://science.sciencemag.org/content/367/6473/83

[132] E. von Holst and H. Mittelstaedt, "The principle of reafference: Interactions between the central nervous system and the peripheral organs," *Perceptual processing: Stimulus equivalence and pattern recognition*, pp. 41–72, 1971.

[133] M. Fukutomi and B. A. Carlson, "A history of corollary discharge: Contributions of mormyrid weakly electric fish," *Frontiers in Integrative Neuroscience*, vol. 14, p. 42, 2020. [Online]. Available: https://www.frontiersin.org/article/10.3389/fnint.2020.00042

[134] Y. Zhang and T. O. Sharpee, "A robust feedforward model of the olfactory system," *PLoS computational biology*, vol. 12, no. 4, p. e1004850, 2016.

[135] J. Cunningham and B. Yu, "Dimensionality reduction for large-scale neural recordings," *Nature Neuroscience*, vol. 17, pp. 1500–1509, 2014.

[136] A. Grinvald and R. Hildesheim, "VSDI: a new era in functional imaging of cortical dynamics," *Nature Reviews Neuroscience*, vol. 5, no. 11, p. 874, 2004.

[137] X.-X. Wei, J. Prentice, and V. Balasubramanian, "A principle of economy predicts the functional architecture of grid cells," *Elife*, vol. 4, p. e08362, 2015.

[138] Z. Goldfeld, E. Van Den Berg, K. Greenewald, I. Melnyk, N. Nguyen, B. Kingsbury, and Y. Polyanskiy, "Estimating information flow in deep neural networks," in *International Conference on Machine Learning*. PMLR, 2019, pp. 2299–2308.

[139] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel, "InfoGAN: Interpretable representation learning by information maximizing generative adversarial nets," in *Advances in Neural Information Processing Systems*, 2016, pp. 2172–2180.

[140] S. Barocas, M. Hardt, and A. Narayanan, *Fairness and Machine Learning*. fairmlbook.org, 2019. [Online]. Available: http://www.fairmlbook.org

[141] T. W. House, "Big data: A report on algorithmic systems, opportunity, and civil rights," 2016. [Online]. Available: https://obamawhitehouse.archives.gov/sites/default/files/microsites/ostp/2016_0504_data_discrimination.pdf

[142] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic attribution for deep networks," in *International Conference on Machine Learning*. PMLR, 2017, pp. 3319–3328.

[143] P. Dabkowski and Y. Gal, "Real time image saliency for black box classifiers," in *Advances in Neural Information Processing Systems*, vol. 30. Curran Associates, Inc., 2017, pp. 6970–6979. [Online]. Available: https://proceedings.neurips.cc/paper/2017/file/0060ef47b12160b9198302ebdb144dcf-Paper.pdf

[144] U. Bhatt, A. Weller, and J. M. F. Moura, "Evaluating and aggregating feature-based model explanations," in *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, C. Bessiere, Ed. International Joint Conferences on Artificial Intelligence Organization, 7 2020, pp. 3016–3022, main track.

[145] P. W. Koh and P. Liang, "Understanding black-box predictions via influence functions," in *International Conference on Machine Learning*. PMLR, 2017, pp. 1885–1894.

[146] B. Kim, O. Koyejo, R. Khanna *et al.*, "Examples are not enough, learn to criticize! Criticism for interpretability." in *NIPS*, 2016, pp. 2280–2288.

[147] C. Olah, A. Satyanarayan, I. Johnson, S. Carter, L. Schubert, K. Ye, and A. Mordvintsev, "The building blocks of interpretability," *Distill*, 2018, https://distill.pub/2018/building-blocks.

[148] C. Olah, A. Mordvintsev, and L. Schubert, "Feature visualization," *Distill*, 2017, https://distill.pub/2017/feature-visualization.

[149] H. Wang, B. Ustun, and F. Calmon, "Repairing without retraining: Avoiding disparate impact with counterfactual distributions," in *International Conference on Machine Learning*. PMLR, 2019, pp. 6618–6627.

[150] H. Wang, H. Hsu, M. Diaz, and F. P. Calmon, "To split or not to split: The impact of disparate treatment in classification," *arXiv preprint arXiv:2002.04788*, 2020.

[151] A. Ghassami, S. Khodadadian, and N. Kiyavash, "Fairness in supervised learning: An information theoretic approach," in *2018 IEEE International Symposium on Information Theory (ISIT)*, 2018, pp. 176–180.

[152] S. Barocas and A. D. Selbst, "Big data's disparate impact," *Calif. L. Rev.*, vol. 104, p. 671, 2016.

[153] J. Angwin, J. Larson, S. Mattu, and L. Kirchner, "Machine bias: There's software used across the country to predict future criminals. and it's biased against blacks." *ProPublica*, 2016. [Online]. Available: https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing

[154] A. G. Ferguson, "Policing predictive policing," *Washington University Law Review*, vol. 94, p. 1109, 2016.

[155] C. Dwork, M. Hardt, T. Pitassi, O. Reingold, and R. Zemel, "Fairness through awareness," in *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, 2012, pp. 214–226.

[156] A. Agarwal, A. Beygelzimer, M. Dudík, J. Langford, and H. Wallach, "A reductions approach to fair classification," in *International Conference on Machine Learning*. PMLR, 2018, pp. 60–69.

[157] D. Dua and C. Graff, "UCI machine learning repository," 2017. [Online]. Available: http://archive.ics.uci.edu/ml

[158] P. Adler, C. Falk, S. A. Friedler, T. Nix, G. Rybeck, C. Scheidegger, B. Smith, and S. Venkatasubramanian, "Auditing black-box models for indirect influence," *Knowledge and Information Systems*, vol. 54, no. 1, pp. 95–122, 2018.

[159] A. Henelius, K. Puolamäki, H. Boström, L. Asker, and P. Papapetrou, "A peek into the black box: exploring classifiers by randomization," *Data mining and knowledge discovery*, vol. 28, no. 5-6, pp. 1503–1529, 2014.

[160] P. Venkatesh, S. Dutta, and P. Grover, "How should we define information flow in neural circuits?" in *2019 IEEE International Symposium on Information Theory (ISIT)*, July 2019, pp. 176–180.

[161] M. J. Kusner, J. Loftus, C. Russell, and R. Silva, "Counterfactual fairness," in *Advances in Neural Information Processing Systems*, 2017, pp. 4066–4076.

[162] C. Russell, M. J. Kusner, J. Loftus, and R. Silva, "When worlds collide: integrating different counterfactual assumptions in fairness," in *Advances in Neural Information Processing Systems*, 2017, pp. 6414–6423.

[163] A. Rényi, "On measures of entropy and information," in *Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Contributions to the Theory of Statistics*. The Regents of the University of California, 1961.

[164] M. Kamiński, M. Ding, W. A. Truccolo, and S. L. Bressler, "Evaluating causal relations in neural systems: Granger causality, directed transfer function and statistical assessment of significance," *Biological Cybernetics*, vol. 85, no. 2, pp. 145–157, Aug 2001. [Online]. Available: https://doi.org/10.1007/s004220000235

[165] K. J. Blinowska, R. Kuś, and M. Kamiński, "Granger causality and information flow in multivariate processes," *Physical Review E*, vol. 70, no. 5, p. 050902, 2004.

[166] M. Dhamala, G. Rangarajan, and M. Ding, "Analyzing information flow in brain networks with nonparametric Granger causality," *NeuroImage*, vol. 41, no. 2, pp. 354–362, 2008.

[167] G. Nolte, A. Ziehe, V. V. Nikulin, A. Schlögl, N. Krämer, T. Brismar, and K.-R. Müller, "Robustly estimating the flow direction of information in complex physical systems," *Physical review letters*, vol. 100, no. 23, p. 234101, 2008.

[168] A. Korzeniewska, M. Mańczak, M. Kamiński, K. J. Blinowska, and S. Kasicki, "Determination of information flow direction among brain structures by a modified directed transfer function (dDTF) method," *Journal of neuroscience methods*, vol. 125, no. 1, pp. 195–207, 2003.

[169] M. B. Schippers, A. Roebroeck, R. Renken, L. Nanetti, and C. Keysers, "Mapping the information flow from one brain to another during gestural communication," *Proceedings of the National Academy of Sciences*, vol. 107, no. 20, pp. 9388–9393, 2010.

[170] A. Brovelli, M. Ding, A. Ledberg, Y. Chen, R. Nakamura, and S. L. Bressler, "Beta oscillations in a large-scale sensorimotor cortical network: directional influences revealed by Granger causality," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 101, no. 26, pp. 9849–9854, 2004.

[171] R. Goebel, A. Roebroeck, D.-S. Kim, and E. Formisano, "Investigating directed cortical interactions in time-resolved fMRI data using vector autoregressive modeling and Granger causality mapping," *Magnetic resonance imaging*, vol. 21, no. 10, pp. 1251–1261, 2003.

[172] G. Deshpande, X. Hu, R. Stilla, and K. Sathian, "Effective connectivity during haptic perception: a study using Granger causality analysis of functional magnetic resonance imaging data," *Neuroimage*, vol. 40, no. 4, pp. 1807–1814, 2008.

[173] C. Bernasconi, A. von Stein, C. Chiang, and P. KoÈnig, "Bi-directional interactions between visual areas in the awake behaving cat," *Neuroreport*, vol. 11, no. 4, pp. 689–692, 2000.

[174] M. Ding, S. L. Bressler, W. Yang, and H. Liang, "Short-window spectral analysis of cortical event-related potentials by adaptive multivariate autoregressive modeling: data preprocessing, model validation, and variability assessment," *Biological cybernetics*, vol. 83, no. 1, pp. 35–45, 2000.

[175] A. Roebroeck, E. Formisano, and R. Goebel, "Mapping directed influence over the brain using Granger causality and fMRI," *Neuroimage*, vol. 25, no. 1, pp. 230–242, 2005.

[176] S. L. Bressler and A. K. Seth, "Wiener–Granger causality: a well established methodology," *Neuroimage*, vol. 58, no. 2, pp. 323–329, 2011.

[177] L. Barnett and A. K. Seth, "The MVGC multivariate Granger causality toolbox: a new approach to Granger-causal inference," *Journal of neuroscience methods*, vol. 223, pp. 50–68, 2014.

[178] M. Ding, Y. Chen, and S. L. Bressler, "17 Granger causality: basic theory and application to neuroscience," *Handbook of time series analysis: recent theoretical developments and applications*, p. 437, 2006.

[179] C. W. Granger, "Investigating causal relations by econometric models and cross-spectral methods," *Econometrica: Journal of the Econometric Society*, pp. 424–438, 1969.

[180] J. Massey, "Causality, feedback and directed information," in *Proc. Int. Symp. Inf. Theory Applic.(ISITA-90)*. Citeseer, 1990, pp. 303–305.

[181] C. J. Quinn, T. P. Coleman, N. Kiyavash, and N. G. Hatsopoulos, "Estimating the directed information to infer causal relationships in ensemble neural spike train recordings," *Journal of computational neuroscience*, vol. 30, no. 1, pp. 17–44, 2011.

[182] W. Hesse, E. Möller, M. Arnold, and B. Schack, "The use of time-variant EEG Granger causality for inspecting directed interdependencies of neural assemblies," *Journal of neuroscience methods*, vol. 124, no. 1, pp. 27–44, 2003.

[183] J. Pearl, *Causality*. Cambridge university press, 2009, pp. 54–57.

[184] J. Schalkwijk and T. Kailath, "A coding scheme for additive noise channels with feedback–i: No bandwidth constraint," *Information Theory, IEEE Transactions on*, vol. 12, no. 2, pp. 172–182, 1966.

[185] C. W. Granger, "Testing for causality: a personal viewpoint," *Journal of Economic Dynamics and control*, vol. 2, pp. 329–352, 1980.

[186] J. Pearl, *Causality*. Cambridge university press, 2009, p. 39.

[187] Y.-H. Kim, A. Lapidoth, and T. Weissman, "The Gaussian channel with noisy feedback," in *Information Theory, IEEE International Symposium on*. IEEE, 2007, pp. 1416–1420.

[188] H. S. Witsenhausen, "A counterexample in stochastic optimum control," *SIAM Journal on Control*, vol. 6, no. 1, pp. 131–147, 1968.

[189] J. Schalkwijk and T. Kailath, "A coding scheme for additive noise channels with feedback–I: No bandwidth constraint," *Information Theory, IEEE Transactions on*, vol. 12, no. 2, pp. 172–182, 1966.

[190] G. Pica, M. Soltanipour, and S. Panzeri, "Using intersection information to map stimulus information transfer within neural networks," *BioSystems*, vol. 185, p. 104028, 2019.

[191] F. Patolsky, B. P. Timko, G. Yu, Y. Fang, A. B. Greytak, G. Zheng, and C. M. Lieber, "Detection, stimulation, and inhibition of neuronal signals with high-density nanowire transistor arrays," *Science*, vol. 313, no. 5790, pp. 1100–1104, 2006.

[192] C. J. Watson, B. J. Venton, and R. T. Kennedy, "In vivo measurements of neurotransmitters by microdialysis sampling," *Analytical Chemistry*, vol. 78, no. 5, pp. 1391–1399, Mar 2006. [Online]. Available: https://doi.org/10.1021/ac0693722

[193] A. Gangrade, P. Venkatesh, B. Nazer, and V. Saligrama, "Efficient near-optimal testing of community changes in balanced stochastic block models," in *Advances in Neural Information Processing Systems*, vol. 32, 2019, pp. 10 364–10 375.

[194] C. E. Carr and M. Konishi, "Axonal delay lines for time measurement in the owl's brainstem," *Proceedings of the National Academy of Sciences*, vol. 85, no. 21, pp. 8311–8315, 1988.

[195] T. I. Netoff, R. Clewley, S. Arno, T. Keck, and J. A. White, "Epilepsy in small-world networks," *Journal of neuroscience*, vol. 24, no. 37, pp. 8075–8083, 2004.